MINIMIZING SHIP MOTIONS
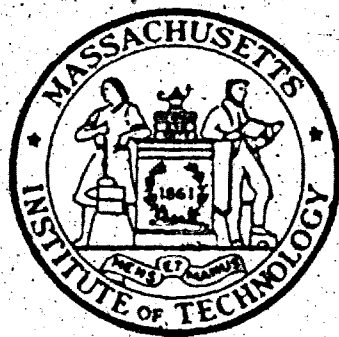
A PRACTICAL APPROACH TO THE OPTIMAL CONTROL OF SYSTEMS WITH APPLICATION TO THE CONTROL OF A SUBMARINE IN A RANDOM SEA

by

DONALD A. GALL
Supervisor E. G. FRANKEL

Contract No. Nonr-1841(90)

Massachusetts Institute of Technology
Department of Naval Architecture and Marine Engineering
Cambridge 39, Massachusetts

Errata Sheet

page 59 and 60 - 2% increments (instead of 1% increments)

page 61   - label on ordinate should read

$$p(\left| x_b / x_a \right| \leq .02)$$

page 86   - Virtual Inertia Coefficients (instead of Virtual Inertial Coefficients)

page 105   - Input disturbance z_D should have a + sign

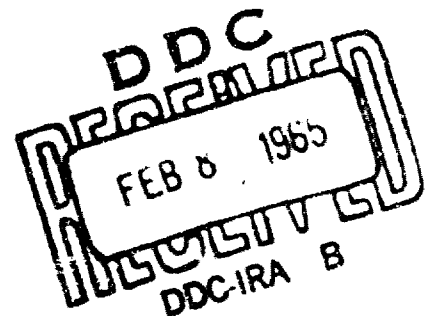MINIMIZING SHIP MOTIONS

A Practical Approach to the Optimal Control of
Systems with Applications to the Control of
a Submarine in a Random Sea

by

Donald A. Gall

Supervisor E. G. Frankel

Contract No.: Nonr 1841(90)
MIT DSR Project No. 9895

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Department of Naval Architecture and Marine Engineering

Report No. 11   August, 1964

ABSTRACT

The basic problem considered is the design of an optimal control for a system subjected to disturbances. Although deterministic disturbances are discussed, the majority of the work and the three examples deal with random or stochastic disturbances. The principal contention of this work is that one cannot determine an "optimal" control for a system without optimizing a performance criterion which is a measure of the performance of the entire system. The Max-Ranking Performance Criterion which is developed here has the capability for considering a wide variety of system variables. The use of this criterion does not depend upon any knowledge of the system interactions; the designer need only present his specifications for the system operation in the form of a ranking array.

The nonlinear systems studied here are analyzed using a statistically equivalent linear system. This follows the work of Kazakov and Booton.

Random search techniques are used with the Max-Ranking Criterion and the linearized analysis approach to produce the optimum system. A simple, but efficient method of random search is developed. Results are obtained to show the expected rate of convergence of this type of search for simple functions.

These methods are then applied to three systems which are subjected to random disturbances: the design of a three-parameter control for a third-order system; the design of a five-parameter control for the roll control of a submarine; and the design of a nine-parameter control for the coupled pitch-heave control of a submarine.

TABLE OF CONTENTS

Table of Figures

Nomenclature

The general nomenclature used throughout this work is listed here. The particular usage, including subscripting, is defined where it is used.

$E(\ )$	expected or mean value
$h(\ )$	system impulse response
J	performance index
K	control parameter
K_{EQ}	statistically equivalent linear gain
m	exponent in the exponential search probability
n	number of search parameters
N_c	number of random choices
p	probability density
$R(\tau)$	autocorrelation function
s	Laplace Transform Operator
$S^2(\omega)$	power spectral density
X, Y	system variables
σ	variance
ω	frequency

1.0 Introduction

One of the principal objectives of a wide variety of control systems is to minimize the detrimental effects which external disturbances have upon the overall performance of the system. In many instances, it is sufficient to design a control which causes the system to recover in some optimal fashion from a step or impulse or other deterministic disturbance. An example of this is the design of a control for a heat exchanger which is subjected to sudden large changes in heat load. In other cases the control must respond optimally to disturbances of a continuous, long-term, random nature. An example of this is the control of the motion of a ship in a random sea.

This work deals primarily with the design of optimal controls for systems subjected to this latter type of continuous, stochastic disturbance. Some aspects of this work are equally applicable to both types of disturbances. Where this occurs, suggestions for the application will be discussed.

The general problem of optimal control is approached here from an overall system point-of-view. It is the principal contention of this work that in order to truly speak of the "optimal control" of a system, it is necessary to optimize a performance criterion which deals in some manner with the performance of the entire system. This approach is quite obvious and straightforward in theory; the difficulty lies in defining a performance criterion which is capable of describing the performance of the entire system. It is felt that the Multiple-Parameter, Max-Ranking Criterion introduced in Chapter 2 is at least a step toward this goal.

The selection of an optimal control is determined through a three-step process:

1. the choice of an optimization criterion,

2. the choice of the mathematical model and analytical approach, and

3. the choice of an optimization technique.

These three operations are treated in Chapters 2, 3, and 4 respectively.

In Chapter 5, the methods arrived at in the previous three chapters are applied to three practical problems. In the first example, a relatively simple, three-parameter, nonlinear control is designed for a second order system subjected to a random input and the results are compared with other methods. In the second, a five-parameter, non-linear controller is designed for the roll control of a submarine subjected to a random sea. Finally, a nine-parameter, nonlinear controller is chosen for the coupled pitch-heave control of a submarine in a random sea.

It is presumed that the reader has a basic knowledge of the response of linear systems to random or stochastic input signals. Many good texts[1, 2, 3] are available in this area and no attempt will be made to duplicate this information in this publication.

2.0 Optimization Criterion

One of the most important steps in any optimization process is the choice of the optimization or performance criterion. It can be safely stated that if the analytical and optimization techniques are adequate, the choice of the optimization criterion will completely determine the final system.

This may appear to be belaboring an obvious point, however one need not look very deeply into the literature in the field of optimal control to see that all too many good analytical efforts are performed on the basis of grossly oversimplified performance criteria. The reason for this is partly historical and partially due to mathematical expediency, as the following sections show.

2.1 Performance Criteria for Randomly Disturbed Systems

When a system is disturbed by a random process, the exact time history of the system output cannot be predicted. However, since some statistical information is known about the input disturbance, it is usually possible to predict certain statistical properties of the output. The two most common and most easily determined statistical quantities are the mean and the variance (or mean square). For non-Gaussian random processes, higher order moments may be of interest. For Gaussian (or normal) probability distributions, the mean and the variance completely describe the distribution. Since this work dealt solely with systems subjected to zero-mean Gaussian disturbances, and since an analytical technique based upon statistically equivalent

- 3 -

linearization will be employed, the variance will provide an effective statistical measure of the system variables for the cases studied here.

2.1.1 Mean Square Error Performance Criteria

In the early 1940's, there existed in the field of communication the problem of extracting as much information as possible from a communications signal which had been distorted by noise. Wiener[4] proposed a method for optimizing this process. His procedure was based upon the   ə of the so-called "minimum mean square error" performance criterion (see Figure 1). Weiner defined an error which was the difference between the actual output signal and the desired output signal (which is, for this particular case, the original signal). He proposed that the optimum system is that system which has the minimum value of the mean square error.

Two questions generally arise concerning the wisdom of the choice of this particular performance criterion.

    1. Why was a statistical measure of the error the only system variable chosen?

    2. Is this the best statistical measure that can be chosen?

The answer to the first question is quite simple. The only objective that was considered important in this phase of the communication system design was to get as much information as possible out of a mixture of signal and noise. The error signal defined above is as effective a measure of the efficiency of this process as can be found.

- 4 -
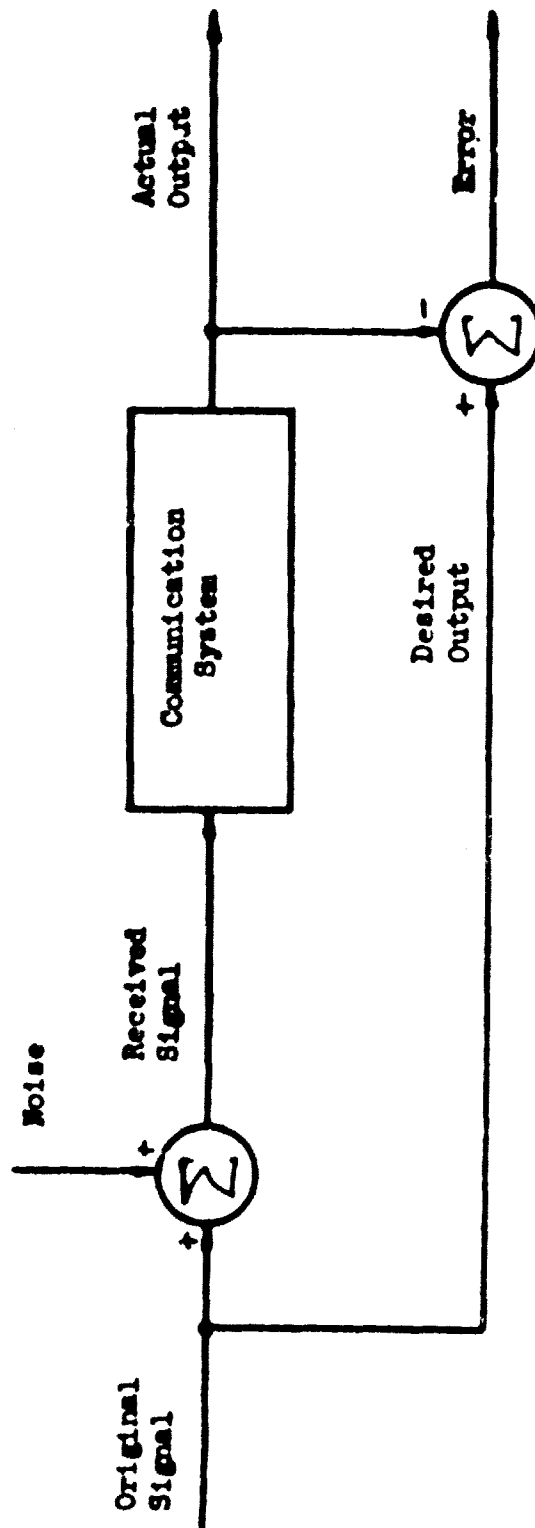
Figure 1

Example of System Error Definition

The answer to the second question took somewhat longer to be formulated. Initially, a number of people suggested that other nonlinear statistical functions of the error would be better. However, in 1958 Sherman[5] showed that most of these more complicated performance criteria would produce a filter identical to that determined using Wiener's method. Sherman stated:

> "Thus, in the case of a Gaussian process,.....
> the solution of the Wiener integral equation (the predictor which minimizes $E(e^2)$) also automatically minimizes $E[\varphi(e)]$ where φ obeys the relationships
>
> $$\varphi(e) = \varphi(-e) \geq 0;$$
>
> $$e_2 \geq e_1 \geq 0 \rightarrow \varphi(e_2) \geq \varphi(e_1)."$$

($E(x)$ is defined as the expected (or mean) value of x.)

Thus we see that for this particular problem the minimum mean square error performance criterion was a good choice.

Wiener's use of the minimum mean square error performance criteria was carried over into the field of automatic control by Phillips[7] and Hall[8]. Phillips suggested a method for determining the coefficients of a linear control system so as to produce a minimum value of the mean square error. He developed a means to integrate the general polynomial form of the power spectral density in order to give the mean square value in closed form. This allows one to employ conventional methods of calculus to obtain a minimum.

In order to consider more effectively what the various performance criteria mean in terms of actual control systems, let us consider

Best Available Copy

the control system of Figure 2. This control is used to position a second order system subjected to a random disturbance. The controller utilizes displacement error (e) feedback. The control force (F) is constrained to within given limits ($\pm F_{max}$), but all other components are considered to be linear. (The Laplace Transform Operator, s , is used to denote differentiation with respect to time.)

Let us first consider a minimum mean square displacement error (σ_e^2) performance criterion. It can be readily shown that the mean square displacement error is zero (and obviously a minimum) if K_D and F_{max} are infinite. This is of very little practical interest, since it implies infinite control forces.

To bypass this difficulty, a number of people have suggested restricting the value of F_{max} to practical values. Now, a minimum value of σ_e is produced by an infinite value of K_D. This required a "bang-bang" or infinite gain, amplitude-limited controller. For the control of Figure 2, it also implies infinite control power. This practical "detail" can be circumvented by placing a constraint on the maximum allowable control power. However, by this time it is obvious that the basic design decisions such as maximum control force and maximum control power are being made without the benefit of the optimization process. The result of this is that one or more optimum coefficients are determined for a control system whose major components are already specified. There is no guarantee that the entire control system is optimal. For this type of problem, we conclude that a minimum mean

- 7 -

Best Available Copy

Figure 2

Simple Positioning Control System

Desired Displacement

Error (e)

Actual Displacement

$$\frac{1}{M s^2 + C s + K_0}$$

K_D

$+F_{max}$

$-F_{max}$

Disturbance

Control Force (F)
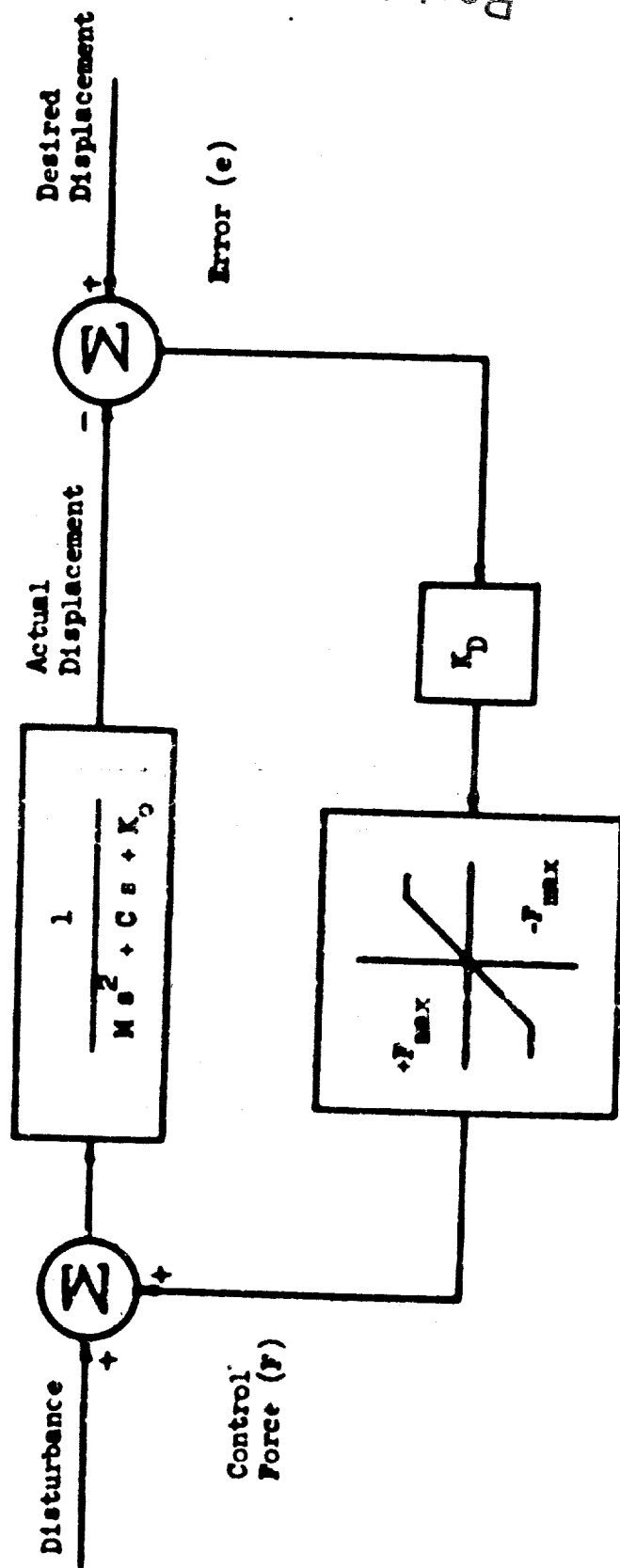
square error performance criterion is not sufficiently general to be of much assistance in the determination of a practical optimal control design.

A number of other criteria which utilize some function of the error have been suggested. References 9, 10, 11, and 12 cover many of these criteria. They will not be discussed here, since they all behave similarly to the minimum mean square error criterion. In fact, for linear systems subjected to Gaussian disturbances, it has been shown[11] that the minimization of the mean square error is equivalent to the minimization of many of these other criteria also.

2.1.2 Multiple-Parameter Weighting Criteria

One step toward the employment of the systems concept in optimization is a direct extension of the work of Phillips[7]. Here, the mean square error performance criterion is replaced by one which includes more of the system characteristics. For example, referring again to the control of Figure 2, it is possible to minimize the function J, where,

$$ J = a_1 \sigma_A^2 + a_2 \sigma_V^2 + a_3 \sigma_e^2 + a_4 \sigma_F^2 \tag{1} $$

The mean square values of acceleration, velocity, displacement error and control force are respectively represented by σ_A^2, σ_V^2, σ_e^2 and σ_F^2. The terms a_1, a_2, a_3, and a_4 are weighting parameters which can be constants or functions of their associated mean square system

- 9 -

variables. If the α's are constants and the system is linear, J can be minimized using standard methods of calculus. It can be shown that increasing K_D and F_{max} tends to decrease σ_A, σ_V, and σ_e while increasing σ_F. Thus if all α's are positive, non-zero and finite, a minimum value of J will exist where σ_A, σ_V, σ_e and σ_F are finite and non-zero. This, at least, is a step in the right direction.

Unfortunately, the process of choosing values for the four α's is quite difficult. An intelligent choice of these weighting parameters must be based upon a thorough knowledge of the system behavior, including the various interactions within the system. This is sometimes possible for simple systems, but it becomes far too difficult as the complexity is increased only slightly.

An alternate approach is the following:

1. make the best possible estimate of the weighting parameters;

2. determine the values of the control parameters which minimize this value of J;

3. determine the mean square values of the necessary system variables;

4. revise the weighting parameters to place more emphasis on those system variables whose mean square values are too high;

5. repeat steps 2 through 4 until a satisfactory system behavior is obtained.

- 10 -

This is certainly not a desirable approach, but it is about
the only one which will give satisfactory results with this method of
multiple-parameter weighting when little is known about the system
behavior ahead of time.

2.1.3 Newton's Multiple Parameter Criteria

Newton[6] recognized the limitations of the minimum mean
square error and suggested that the mean square error should be min-
imized while simultaneously limiting the mean square values of other
important system variables (such as control forces, etc.). This
requires introducing one or more Lagrange Multipliers. If the con-
trol configuration is known, the n control parameters can be
determined from the n equations of the form,

$$\frac{d}{dK_i}\left[\sigma_e^2(K_1\ldots K_i\ldots K_n) + \sum_{j=1}^{m} \lambda_j \varphi_j(K_1\ldots K_i\ldots K_n)\right] = 0 \quad (2)$$

where $\sigma_e^2(K_1\ldots K_i\ldots K_n)$ is the mean square error expressed as a function
of the n control parameters $(K_1\ldots K_i\ldots K_n)$, λ is the Lagrange Multi-
plier and the m constraints are expressed by the m functions,
$\varphi_j(K_1\ldots K_i\ldots K_n) = 0$. This, however, is exactly equivalent to the
multiple-parameter, weighting criterion (equation 1) with α_3 equal to
unity. The remaining weighting parameters become constants equal to the
Lagrange Multipliers of equation (2). Thus, this method has the same
deficiencies as the multiple-parameter weighting criterion.

Newton went beyond this point and chose to leave the control configuration unspecified. He used Calculus of Variations techniques with the Lagrange Multipliers to find the optimum linear control function. This approach is certainly superior to any methods thus far discussed, since it determines both the control configuration and the control parameters. However, it does not solve the problem of choosing the weighting functions or Lagrange Multipliers.

2.1.4 Multiple Parameter Max-Ranking Criterion

The method described in this section is based upon the employment of a ranking array. The essential factor here is that each system attribute which is to be considered in the optimization is rated against an absolute scale of desirability. An example of a possible ranking array for the system of Figure 2 is shown below.

Desirability	$J(i)$	$(i=1)$ σ_A (ft/sec^2)	$(i=2)$ σ_V (ft/sec)	$(i=3)$ σ_e (ft)	$(i=4)$ σ_F (lb_f)
Most Desirable	0	0	0	0	0
	1	3	8	0.1	10
	2	3	10	0.2	20
	3	4	12	0.3	40
Least Desirable	4	8	16	0.4	80

Table 1

Sample Ranking Array

The most desirable values of the root mean square (r.m.s.) acceleration, velocity, displacement error, and control force are all zero. This is, of course, an impossible goal as long as there is any disturbance force. The final row of the table can be readily filled in also. This row represents the maximum value of each of the system variables which is acceptable.

The remainder of the table is constructed with as many rows as the designer feels are necessary to adequately describe the various levels of desirability of the system response. The only external restriction placed on the table's construction is that each column must be either monotonically non-increasing or monotonically non-decreasing. Each r.m.s. value is chosen to be as desirable as all other values in the same row. For example, consider row 3 ($J(1) = 2$). A value of $\sigma_A = 3$ ft/sec^2 is considered as being as desirable an end result as $\sigma_V = 10$ ft/sec, and $\sigma_e = 0.2$ ft., and $\sigma_F = 20$ lb$_f$.

The construction of this ranking array should be carried out with a great deal of thought. The results of the entire optimization study will depend upon the values selected at this time. In order to construct the ranking array, the designer must have a good appreciation of the system capabilities and requirements. However, he need not be familiar with how the various system variables are related to each other and to each of the several control parameters. This function is taken care of automatically by the analytical and optimization studies which are conducted in the following phases of this work.

Once the ranking array is set up, it can be applied in a straightforward and simple manner. The first step is to assign values to each $J(i)$ for any given set of system variables. This can be accomplished by any interpolation scheme which the designer desires to employ.

As an example, let us refer back to Table 1. Let us suppose that for a given system with a given disturbance a certain set of control parameters gives the following results:

$$\sigma_A = 4.0 \text{ ft/sec}^2$$

$$\sigma_V = 4.0 \text{ ft/sec}$$

$$\sigma_e = 0.15 \text{ ft}$$

$$\sigma_F = 20.0 \text{ lb}_f$$

Linear interpolation in Table 1 gives the following values of each $J(i)$:

$$J(1) = 3.0 \text{ (corresponding to } \sigma_A = 4.0 \text{ ft/sec}^2)$$

$$J(2) = 0.5 \text{ (corresponding to } \sigma_V = 4.0 \text{ ft/sec)}$$

$$J(3) = 1.5 \text{ (corresponding to } \sigma_e = 0.15 \text{ ft)}$$

$$J(4) = 2.0 \text{ (corresponding to } \sigma_F = 20.0 \text{ lb}_f)$$

This essentially established the desirability (for this particular case) of each of the four resulting system variables. Two different approaches were taken in an attempt at assigning an overall system desirability based upon these separate values of the individual $J(i)$'s.

The first method defines the overall system desirability as the average desirability of the resulting individual system variables. For the particular example being considered, this gives:

$$J_s = \frac{1}{4} \sum_{i=1}^{4} J(i) = \frac{1}{4}(3.0 + 0.5 + 1.5 + 2.0) = 1.75$$

This method is exactly equivalent to the multiple-parameter weighting criterion (equation (1)) whose weighting parameters are functions determined by the ranking array. This method has only one advantage over the weighting method, a methodology for determining the weighting functions. The principal disadvantage is that there is an implicit weighting between the columns of the ranking array. To illustrate this, consider the system of Figure 2, where increasing K_D will cause an increase in σ_y while decreasing σ_A, σ_V and σ_e. The minimum value of the system desirability (J_s) based upon this averaging method would be one which produced low values of $J(1)$, $J(2)$, and $J(3)$ at the expense of higher values of $J(4)$.

A much better method is one which equates the overall system desirability with the value of $J(i)$ corresponding to the least desirable of the resulting individual attributes of the system. This, in effect, states that the system is no more desirable than its least desirable attribute. This method has been termed Max-Ranking. The Max-Ranking measure of system desirability (J_M) is defined simply as:

$$J_M = \left[J(i)\right]_{max}$$

Thus, for the example being used here, J_M is,

$$J_M = \left[J(1) = 3.0; \; J(2) = 0.5; \; J(3) = 1.5; \; J(4) = 2.0\right]_{max}$$

or

$$J_M = J(1) = 3.0$$

The criterion of optimality is that J_M should be minimized. This tends to produce a final system with two or sometimes more of the values of the individual $J(i)$'s being equal.

The Max-Ranking method has several advantages over other performance criteria. First, the desired response of the various system variables can be specified in a form which is familiar to the designer. This is simply because the designer will (or should) have an awareness of what value of r.m.s. displacement error or acceleration or control force would be considered good or poor for his particular application.

Secondly, there is no restriction on the number of system variables which can be considered in the ranking array. Any variables included in the ranking array which do not turn out to limit the final design are automatically excluded in the optimization.

For some types of optimization techniques, the computation time can be reduced compared to some other multiple parameter methods.

This is because each individual $J(i)$ can be compared to the previous minimum value. As soon as a value of $J(i)$ is found which exceeds the previous minimum, the control parameter set can be discarded. There is no reason to calculate all the remaining values as would be required for a multiple-parameter weighting criterion.

The Max-Ranking method has one disadvantage for small simple systems. Because of the possible nonlinearity of the ranking array and the inherent nonlinearity of the maximum selection, it is almost always impossible to obtain a closed-form algebraic solution for J_M. This, coupled with the fact that the several partial derivatives of J_M taken at the minimum point are, in general, discontinuous usually disallows any way of minimizing J_M based upon calculus. The discontinuities in the partial derivatives also add to this to make steepest ascent techniques quite difficult to apply.

This, at first, appears to be a severe disadvantage. However, optimization techniques based upon calculus and steepest ascent become very unwieldy anyhow as one attempts to optimize large order systems which have several control parameters. Chapter 4 deals with this problem of choosing an optimization technique and reaches the conclusion that for larger systems, the Max-Ranking method is actually simpler to work with than other Multiple-Parameter methods.

The Max-Ranking method is applied in each of the examples in Chapter 5. Some additional practical guidance on the use of this method is discussed at that point.

2 2 Performance Criteria for Systems with Deterministic Disturbances

This topic will only be dealt with briefly, since Gibson[12,13] and Wolkovitch[14] have already covered most of the standard criteria quite effectively. What will be discussed is the fact that the Multiple-Parameter, Max-Ranking Method can also be applied to a system subjected to deterministic disturbances.

Let us consider again the simple positioning control system of Figure 2. However, this time we will assume that the disturbance is deterministic and of short duration. (As a matter of fact, the system disturbance could be a step, ramp or other known change in the desired displacement.) Some measures of system response which are of value for this case include:

1. integral square error

2. integral time averaged error

3. settling time

4. percent overshoot

5. maximum control force

6. maximum control power

7. maximum acceleration.

These or other meaningful parameters can be used for the several headings of the ranking array. Numbers which describe the desirability of these parameters for a particular disturbance would then be filled in as indicated before in Section 2.1.4. Optimization methods which are described in Chapter 4 can be employed to optimize the value of overall system desirability (J_M).

A further extension of this is to consider a system which is subjected to a continuous, random disturbance while attempting to follow deterministic changes in the desired displacement. If a case existed where the specification on both of these requirements were restrictive, a joint Max-Ranking Array can be constructed which contains both stochastic and deterministic measures of system response. For each set of control parameters, the response to both the random disturbance and the deterministic command input can be calculated and an overall system desirability based on the two calculations.

3.0 Analytical Techniques

This chapter deals with the problem of determining the response of a particular type of nonlinear system to a stochastic disturbance. The nonlinearity which is considered is the symmetric, unity-gain, unity-amplitude, saturating element whose output (Y) can be described by the following function of the input (X):

$$Y = \begin{cases} +1 & \text{for } X \geq +1 \\ X & \text{for } |X| \leq +1 \\ -1 & \text{for } X \leq -1 \end{cases} \qquad (3)$$

The general form of the symmetric saturating element can be expressed by adding two linear gain terms to the basic unity saturating element as shown in Figure 3. The upper and lower saturating limits are $\pm K_2$ and the gain in the central linear region is the product of K_1 and K_2.

Two approaches to the general problem are discussed. Section 3.1 deals with direct analog and digital simulation and Section 3.2 considers the problem from the point-of-view of statistically equivalent linearization.

3.1 Simulation Techniques

Simulation methods are usually used for complex or nonlinear systems which cannot be handled readily by the more easily applied linear techniques. For the type of problem that is being studied here, there are two additional factors which must be considered. First,

Figure 3

General Form of the Symmetric Saturating Element
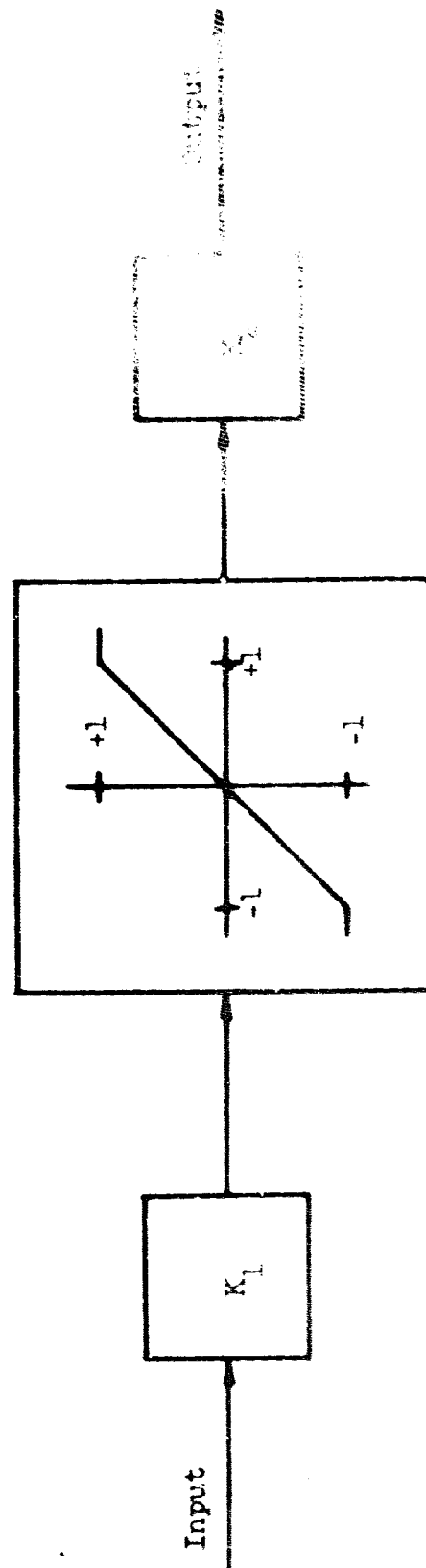
since the input disturbance is stochastic, it will be necessary to run the simulation for a time which will be long enough to assure accurate mean square values of the system variables. Secondly, it may be necess to test a large number of control parameter combinations in order to determine the optimum set.

The use of analog simulation was ruled out because of the lengthy time required to find an optimum. With the available facilitie individual run times of the order of a few minutes would be the best that could be expected. The fact that several thousand of these individual runs are required for three or four parameter optimization problems leads to the inevitable conclusion that much higher speed analo equipment is required for this type of work.

On the other hand, the very requirements which caused the rejection of the use of analog simulation favor the use of high speed digital calculation. Present day high speed digital computers have no trouble simulating complex, nonlinear systems. The major problems of this method are the choice of the type and length of the random input disturbance and the choice of the computing increment to be used in the calculation.

This method shows sufficient potential to merit is retention at this time. It is compared with linearized methods in Section 3.3.

3.2 Linearized Approach

A second approach to the problem is to replace the nonlinear element by a linear element which approximates the behavior of the non-linearity. This allows the problem to be solved by conventional linear techniques.

For systems subjected to deterministic disturbances, the describing function[13, 15] can be used. Stochastically disturbed systems, necessitate the use of a method of linearization that produces a linear element which is, in some fashion, statistically equivalent to the original nonlinearity. The problem was studied independently by Kazakov[16] and Booton[17, 18]. Kazakov stated the general problem of finding the unknown moments of the probability distribution at the output of the nonlinearity as functions of the moments at the input. For a linear system this is relatively simple, since the probability moments of the random functions are linearly transformed; the nth order output moment is dependent only upon the nth order input moment. For the nonlinear case, the nth order output moment is also a function of the lower and higher order moments of the input. If the input to the nonlinearity has a Gaussian (or normal) probability distribution, the first two moments (mean and variance) completely describe this distribution. Thus, all output moments are completely determined by these two input moments.

Kazakov defines three means of linearization. In the first, linear constants are chosen which produce the same values of the first

and second moments as are produced by the nonlinearity. The second method was also suggested by Booton and, in the United States, is usually referred to as Booton's method. This approach minimizes the mean square difference between the output of the linear and nonlinear elements. The third method of Kazakov simply averages the results of the first two linearization methods. Kazakov[16] lists these equivalent gains for a wide variety of static nonlinearities which are subjected to a Gaussian input.

For the symmetric, unity-gain, unity-amplitude, saturating element of equation (3), which is subjected to a zero mean, Gaussian input with variance, σ_X, the equivalent gain calculated by the first method (K_{EQ1}) can be shown to be:

$$K_{EQ1} = \frac{1}{\sigma_X} \{ 1 + (\sigma_X^2 - 1) \; erf \left(\frac{1}{\sqrt{2} \; \sigma_X} \right)$$

$$- \sqrt{\frac{2}{\pi}} \; \sigma_X e^{-\frac{1}{2\sigma_X^2}} \}^{\frac{1}{2}}$$

(4)

The second approach (Booton's Method) produces,

$$K_{EQ2} = erf \left(\frac{1}{\sqrt{2} \; \sigma_X} \right)$$

(5)

The third method gives,

$$K_{EQ3} = \tfrac{1}{2}(K_{EQ1} + K_{EQ2})$$ (6)

These three functions are plotted in Figure 4. Kazakov shows that, for a Gaussian input, K_{EQ1} is the upper limit and K_{EQ2} is the lower limit on the actual equivalent gain, with K_{EQ3} approximating the relationship best over the full range of σ.

When nonlinearities occur within feedback control systems, there is no guarantee that the input to the nonlinearity is Gaussian, since the non-Gaussian output from the nonlinearity is fed back through the system to the input. Originally, it was felt that the application would be limited to nonlinearities which are only "slightly" nonlinear so that the Gaussian probability distribution would not be measurably altered. Subsequently, it was observed that even severely distorted signals were again Gaussian after passing through, for example, two simple low pass filters of the form $1/(s + \alpha)$.

The reason for this is best explained by considering time domain analysis[19]. The response of a system, $y(t)$, can be calculated from the convolution (or superposition) integral,

$$y(t) = \int_{-\infty}^{t} h(t - \tau)\, X(\tau)\, d\tau$$ (7)

where $h(t - \tau)$ is the system response to an impulse at time $t = \tau$, and $X(\tau)$ is the input signal.

Figure 4

Equivalent Linearization

of the Saturating Element
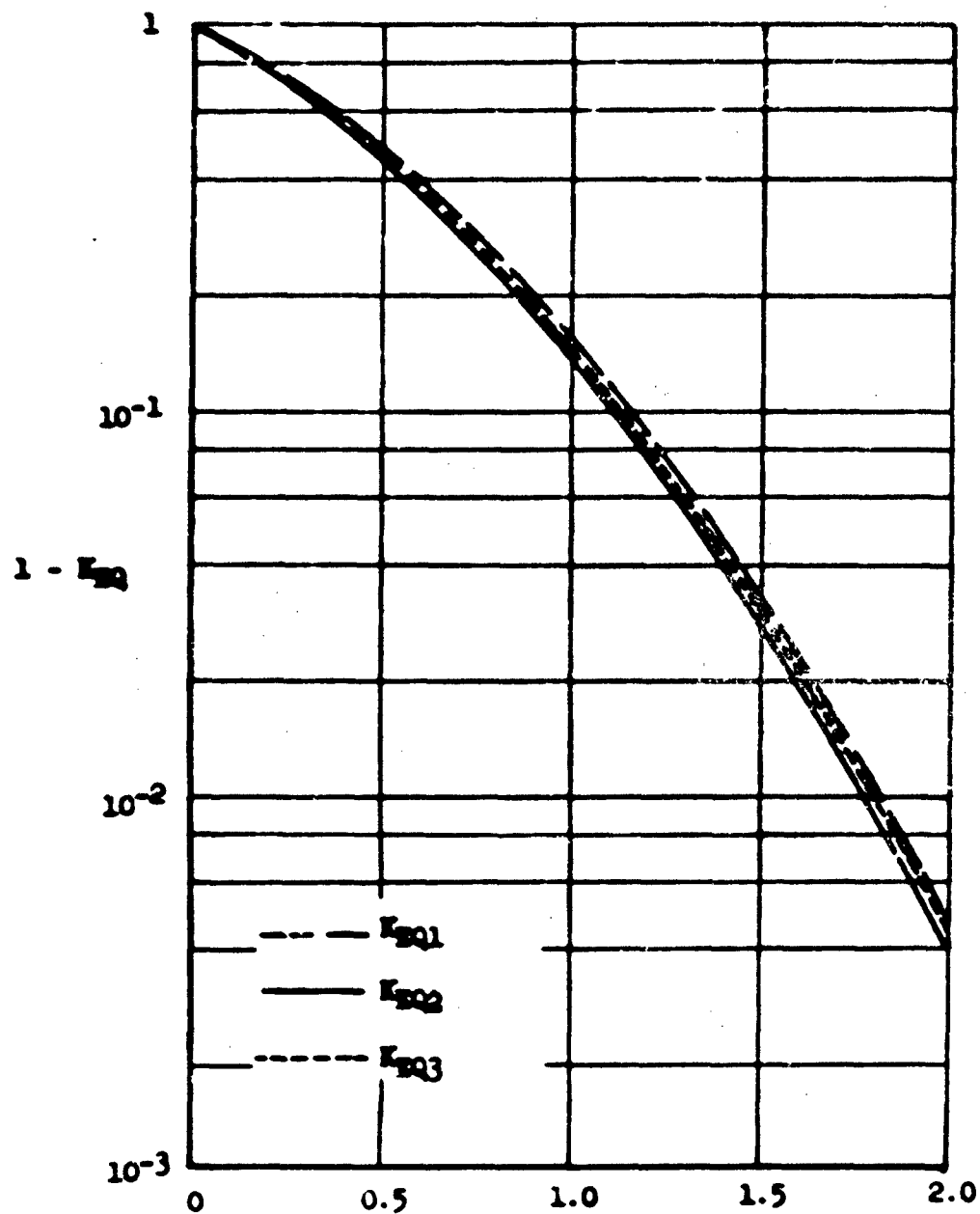
This integral "weights" past values of $X(\tau)$ according to $h(t - \tau)$ and then sums these results over all past history. In his sense, it is analogous to weighting and summing a large number of individual signals, which, according to the Central Limit Theorem, should produce a Gaussian probability distribution.

The degree to which the output approaches a Gaussian distribution depends upon the type of system. The impulse response of a simple gain is an inpulse at time $t = \tau$. Thus, this convolution integral only samples $X(\tau)$ at one period of time so that the probability distribution form remains unchanged. Conversely, a simple integration gives a uniform impulse response for all τ less that t. This convolution samples all previous values of $X(\tau)$ uniformly, and as a consequence always produces a pure Gaussian output probability distribution. In between these two extremes different types of filters produce a varying tendency for a return to Gaussian distribution. A simple lag with its impulse response,

$$h(t - \tau) = e^{-\alpha(t - \tau)} \qquad \text{for } \tau \leq t$$

$$(8)$$

$$h(t - \tau) = 0 \qquad \tau > t,$$

samples primarily in the vicinity of $\tau = t$ and therefore produces a smaller restoration of the Gaussian distribution than, for instance, two simple lags. Two equal, first-order lags have an inpulse response

of,

$$h(t - \tau) = (t - \tau)e^{-(t - \tau)} \qquad (9)$$

which produces a wider sampling displaced somewhat from $\tau = t$. Therefore, a qualitative measure of the tendency for returning to a Gaussian probability distribution can be obtained by looking at the impulse response of the filter.

One other problem exists which is peculiar to this type of optimization study. One of the end results of optimization will very likely be the specification of the optimum linear gain for the general saturating nonlinearity shown in Figure 5. Since only the values of σ_{X_a}, K_3 and σ_{X_d} will be known, it is necessary to work backwards to determine K_1, K_{BQ} and K_2.

Figure 5 shows the steps involved in linearizing the general limiting element. Figure 4 expresses the relationship between the three different, "equivalent" linear gains and the parameter β,

$$\beta = \frac{1}{\sqrt{2}\,\sigma_{X_b}} \qquad (10)$$

Since σ_{X_b} will not, in general, be known, it is necessary to express these equivalent gains in terms of σ_{X_a}. If we define a parameter γ equal to,

$$\gamma = \frac{K_2}{\sqrt{2}\,K_3\,\sigma_{X_a}} \qquad (11)$$

- 28 -

Figure 3

Steps in Linearizing the General Saturating Element
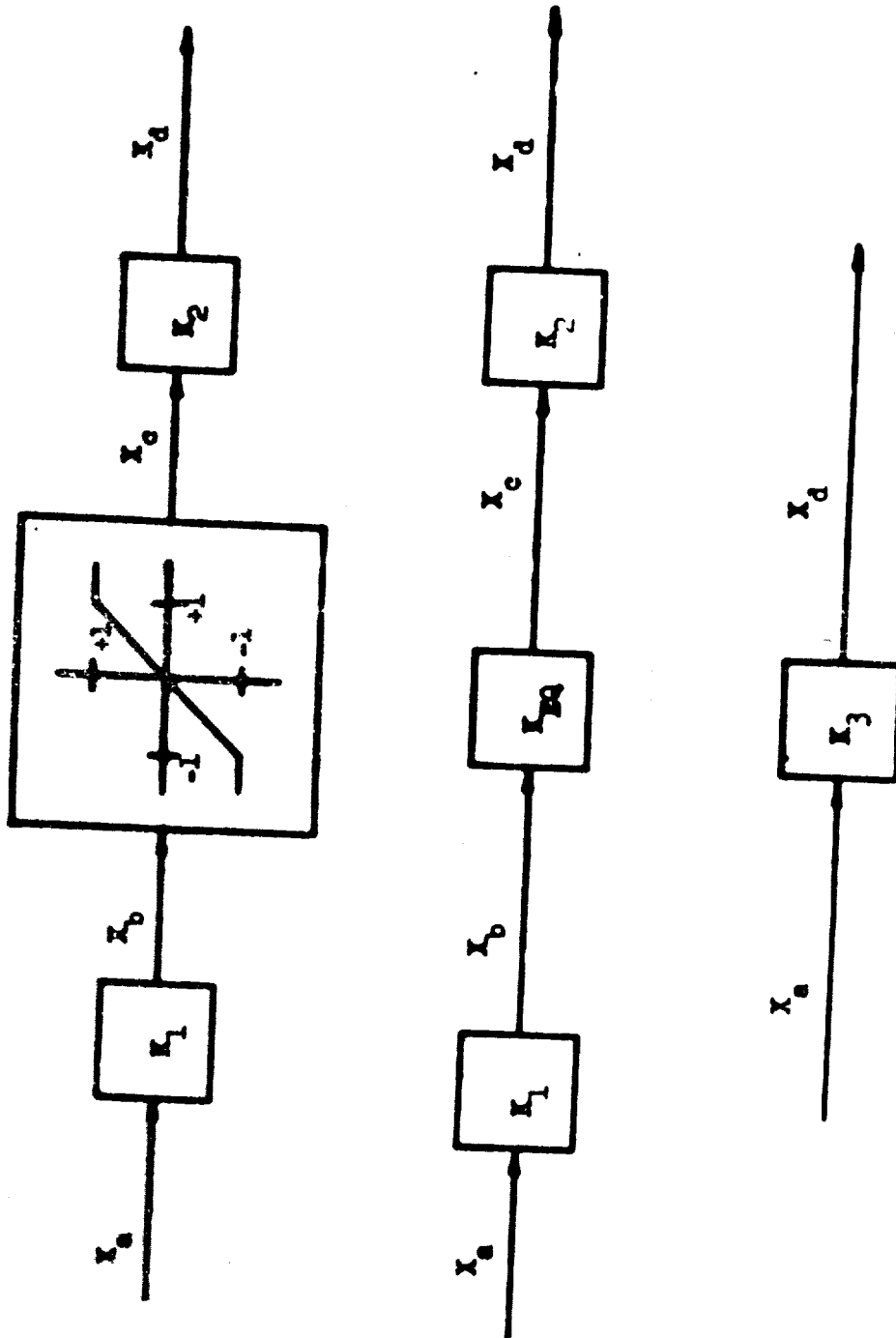
we see that,

$$\beta = K_{EQ} \gamma \tag{12}$$

This now allows us to calculate the   equivalent gains as a

function of γ, by using Figure 4. These results are plotted in Figure

6. Since K_3 and σ_{X_a} will be known from the optimization results, the

equivalent gain can now be determined directly from the saturation

limit, K_2.

It is interesting to note that each curve of Figure 6 has a

minimum value of γ for which a value of K_{EQ} exists. For the first,

equal variance method, this value of γ is $\sqrt{2}/2$ It is readily seen

that this corresponds to a value of σ_{X_c} equal to unity. Since the

maximum value that X_c can attain is unity, it is obvious that its

r.m.s. value can never exceed unity. The second method gives this

limiting value of γ as $\sqrt{\pi}/2$. This corresponds to a maximum value of

σ_{X_c} of $\sqrt{2/\pi}$. This points out the fact that the mean square output

of the linear and nonlinear elements will not, in general, be equal for

this method. This difference, of course, originates from the assumption

basing statistical equivalence on a minimum mean square error. For this

type of nonlinearity, this difference can be almost totally attribut d

to the large fourth order probability moment which is present in the

nonlinearity output. This component is ignored by the first, equal

variance, method.

3.3 Comparison of Simulation and Linearization Techniques

Two aspects of the employment of these two approaches will be

discussed here. The first deals with how readily each method may be

Figure 6

Equivalent Linearization

of the Saturating Element
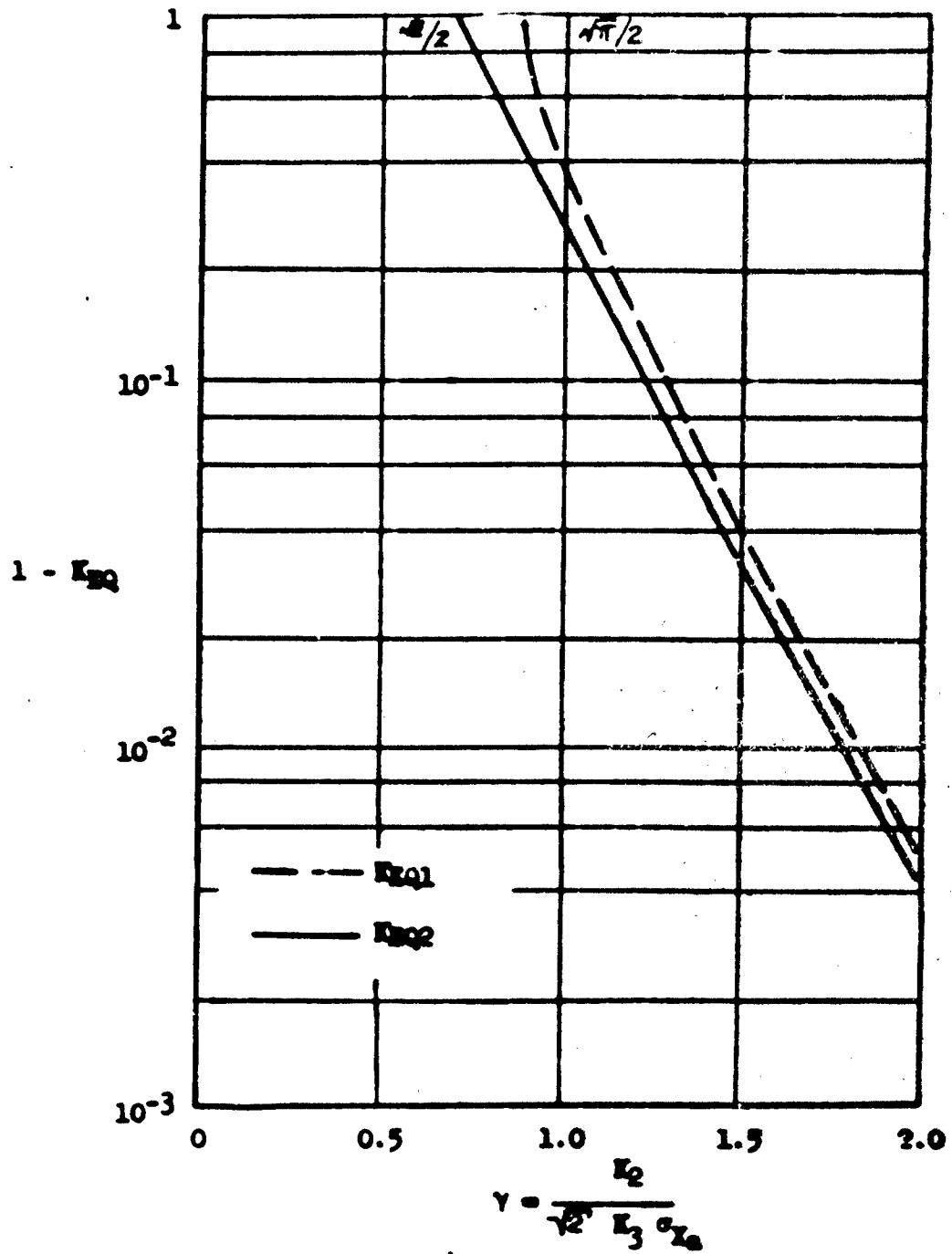
adapted for digital computer calculation. The second is concerned
with a comparison of the computation time required by the two methods.

3.3.1 Adaptability to Digital Computer Calculation

Any method of digital computer simulation requires the use
of some type of finite difference calculation method. One of the more
critical steps with finite difference approaches is the choice of the
time increment to be used in the calculation. Time intervals which are
too long lead to inaccuracies or even completely erroneous results. If
the time interval chosen is shorter than is necessary, long computation
times will result.

In optimization studies it is sometimes necessary to vary the
control parameters over wide ranges. This can cause the system's
natural frequencies to vary widely and result in the additional require
ment that a satisfactory value of the time interval be determined for
each set of control parameters which is tried. This complicates the
computer program and can add a significant amount of computation time
to the solution of a problem.

If a linearized solution is used, finite difference techniques
may not be required, Appendix E of Newton, Gould, and Kaiser[20] lists
the general algebraic solutions of the mean square values for first
through tenth order systems. These are in a form which is readily
adaptable to digital computer calculation. If these equations are used,
two very important facts must be remembered:

1. These equations do not apply to unstable systems. Therefore, stability must be checked by means of the Routh or other criterion before using them.

2. Round-off error can be a problem with the higher order systems. It is generally recommended that the equations for seventh through tenth order systems should not be used unless double accuracy (16 place) computer calculation methods are employed[21].

For high order systems, it may be simpler and quicker to use finite difference integration of the general integral,

$$\sigma_X^2 = \frac{1}{2\pi j} \int_{-j\infty}^{+j\infty} \frac{c(s) \; c(-s)}{d(s) \; d(-s)} \; ds \qquad (13)$$

than to employ double accuracy calculation methods. Direct integration is particularly simple if the input disturbance has a narrow band power density spectrum This permits equation (13) to be reduced to:

$$\sigma_X^2 = \frac{1}{\pi} \int_{L_1}^{L_2} \frac{c(j\omega) \; c(-j\omega)}{d(j\omega) \; d(-j\omega)} \; d\omega \qquad (14)$$

where L_1 and L_2 represent the lower and upper frequency limits of the narrow band power density spectrum of the system input.

If, however, the input spectrum is considerably wider than the expected frequency response of the system tranfer function, the problem is more difficult. In this case, the peaks of the output

- 33 -

power spectrum may vary considerably due to changes in the control parameters. One now has the choice of using finite difference integration with a small frequency increment over a wide range of frequencies or tailoring the finite difference integration to match each particular output power spectrum. The first method results in longer computation times, while the second requires a more complex computer program.

3.3.2 Computation Time Requirements

Although it is rather ridiculous to speak of the computation time for an "average" n^{th} order system, it is necessary to have some way of comparing the several analytical methods to determine which can give results in a reasonable period of time. For this reason, some estimates of computation time for the IBM 7094 computer have been made based on the following assumptions:

1. The time required to determine the coefficients of the nth order transfer function is equal to the time that it takes to evaluate the corresponding n^{th} order equation of Appendix E of Newton, Gould, and Kaiser[20]. (This turns out to be reasonably accurate for the 3rd, 6th, and 10th order systems studied in Chapter 5.)

2. Finite difference integration of the power spectrum will require an average of fifty points.

3. The computation time of simulation methods is proportional to the order of the system.

The results for the algebraic evaluation and integration methods were obtained for a single mean square value (m = 1) and for the calculation of n mean square values (m = n).

The simulation approach automatically gives all mean square values since it was based upon a simple trapezoidal finite difference integration. Some time saving might be made by using larger time increments with a more accurate simulation method such as Runge-Kutta[22] or the Matrix Exponential[23, 24]. The initial results were sufficiently poor for systems of tenth order and less to discourage any additional work along these lines at this time.

In spite of the estimations involved in obtaining these approximate relations, a few conclusions can be drawn from Figure 7:

1. Direct algebraic evaluation is superior for systems up to the point where double accuracy calculations are required.

2. In the neighborhood of tenth order systems, either integration or double accuracy algebraic evaluation can be used. The decision should be based upon the number of integration points to be used, the average number of mean square values to be calculated, etc. (It should be remembered that the integration method is easier to program than the algebraic evaluation method for systems of this order.)

Figure 7

Estimated Calculation Times

For Three Analytical Methods
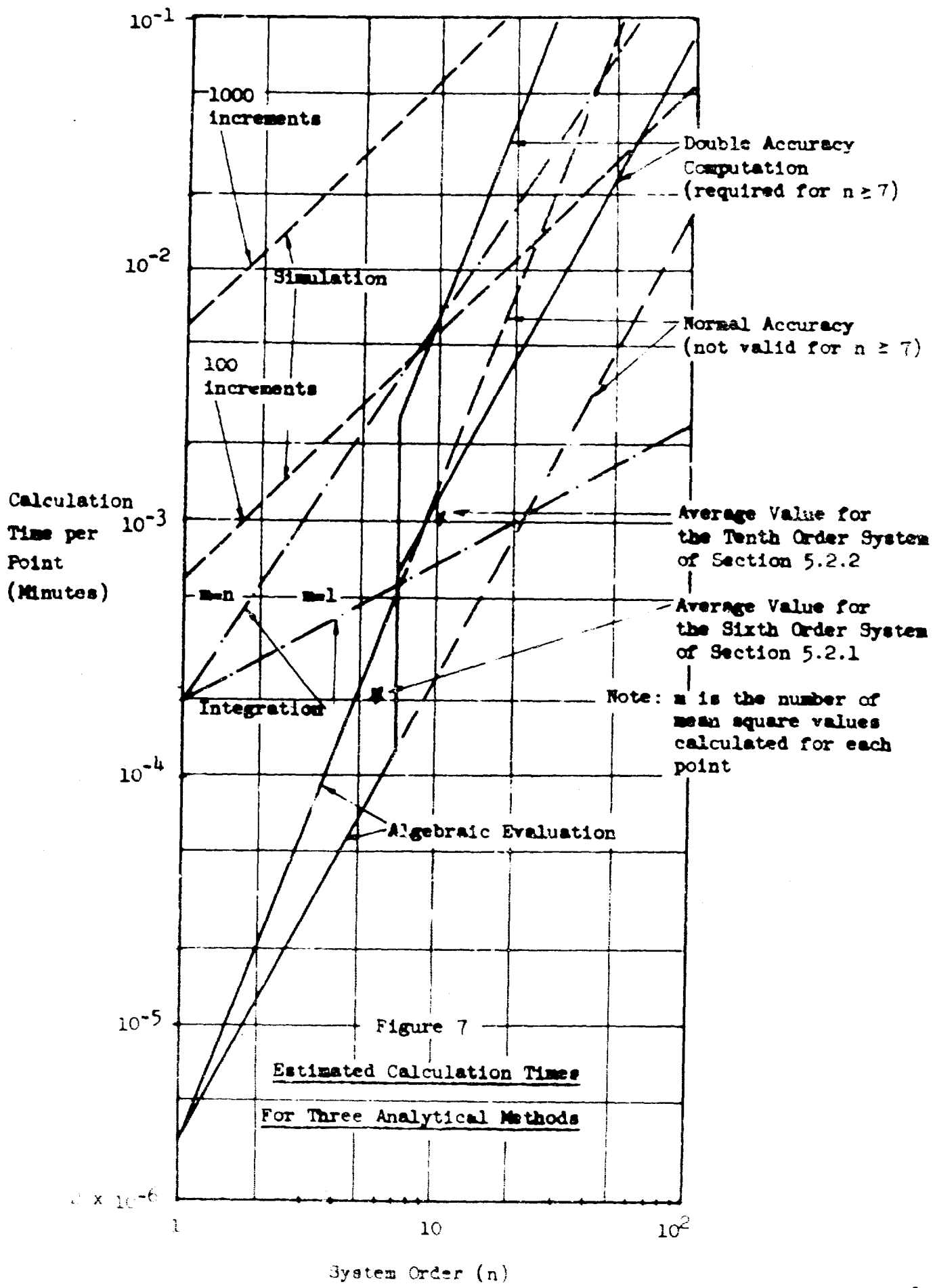
3. Direct integration should be used for systems above fifteenth or twentieth order.

4 Simulation should only be used when the program simplicity is more important than the computation time, or when the system nonlinearities cannot be handled properly through equivalent linearization.

4.0 Optimization Techniques

The choice of the Multiple-Parameter, Max-Ranking Optimization Criterion which was introduced in Chapter 2 places a restriction on the type of optimization technique which can be employed. Standard methods of calculus cannot be used because the slopes at the various minima will not, in general, be zero. The reasons for this are given in Section 2.1.4 and examples are shown in Figure 8 and in Sections 5.2 and 5.3.

Some types of steepest ascent[25] optimization might be employed. The general n-dimensional steepest ascent method uses some form of the following logic:

1. Determine the partial derivatives with respect
    to each of the n dimensions at the present position.
    This is usually done by calculating the value of the
    function whose minimum is being sought at small
    increments on either side of the present position.
    The average partial derivatives through the present
    location can then be calculated for that particular
    dimension. This is repeated for all n dimensions.

2. Determine a new location by choosing increments
    for each of the n dimensions proportional to their
    own partial derivatives. (The directions chosen
    for each dimension, of course, depends upon
    whether a maximum or a minimum is being sought.)

- 38 -

This particular choice of increments forces
the new location to be in the direction of the
steepest path which passes through the old
location.

3. Steps 1 and 2 are repeated until a local max-
imm (or minimum) is reached. In actuality,
the search is stopped when all the partial der-
ivatives arc below some predetermined level,
since otherwise the search could go on indefin-
itely looking for exactly zero partial derivatives.

4. Repeat steps 1 through 3 for several starting
points. This is necessary, since each starting
point can only result in the determination of a
single extremum. In most multidimensional
problems there will be several relative extrema.

The introduction of the Max-Ranking Performance Criterion
creates two additional problems for the steepest ascent method. Since
the surface whose minimum is being sought is a composite formed by the
upper selection of a number of surfaces, there will be intersections
of the surfaces with corresponding discontinuous derivatives. This
can cause problems in the determination of the local partial deriva-
tives and in the decision as to when a local extrema has been reached.
Secondly, most extrema will occur at the intersection of two or more
of these hypersurfaces. Thus it will be certainly possible to have

large valued derivatives in the immediate vicinity of an extremum. Figure 8 shows a typical performance curve for a one-dimensional optimization which is employing the Max-Ranking Criterion. A high value of the slope at position (a) would dictate a large incrementing step (ΔK) to location (b). This could very easily result in a search process which oscillated back and forth without ever converging on the actual minimum unless special precautions were taken to guard against this situation. These difficulties add complexity to an optimization procedure which, for multiple dimensional searches, is already quite complex and lengthy.

The steepest ascent method provides excellent results for a search of well-behaved, unimodal functions. Brooks[26] has shown that for unimodal functions of two variables the steepest ascent method is superior to a number of other methods. In particular, Brooks states that sequential methods (i.e., methods which base a course of action, at least in part, upon the past results) give generally better results than non-sequential methods. However, if it is necessary to find the extrema of functions which are not unimodal, the problem is more complicated. It is now necessary to use a number of starting points to be assured that all local extrema in the range of interest are found.

If the required number of starting points for a one-dimensional problem is "M", the number of starting points for the equivalent n-dimensional problem is M^n. This can be reduced somewhat by various additions to the computer program. For example, since a large number of starting points may result in determining the same extrema, the

- 40 -

Figure 8

Typical One-Dimensional Max-Banting Performance Index Curve
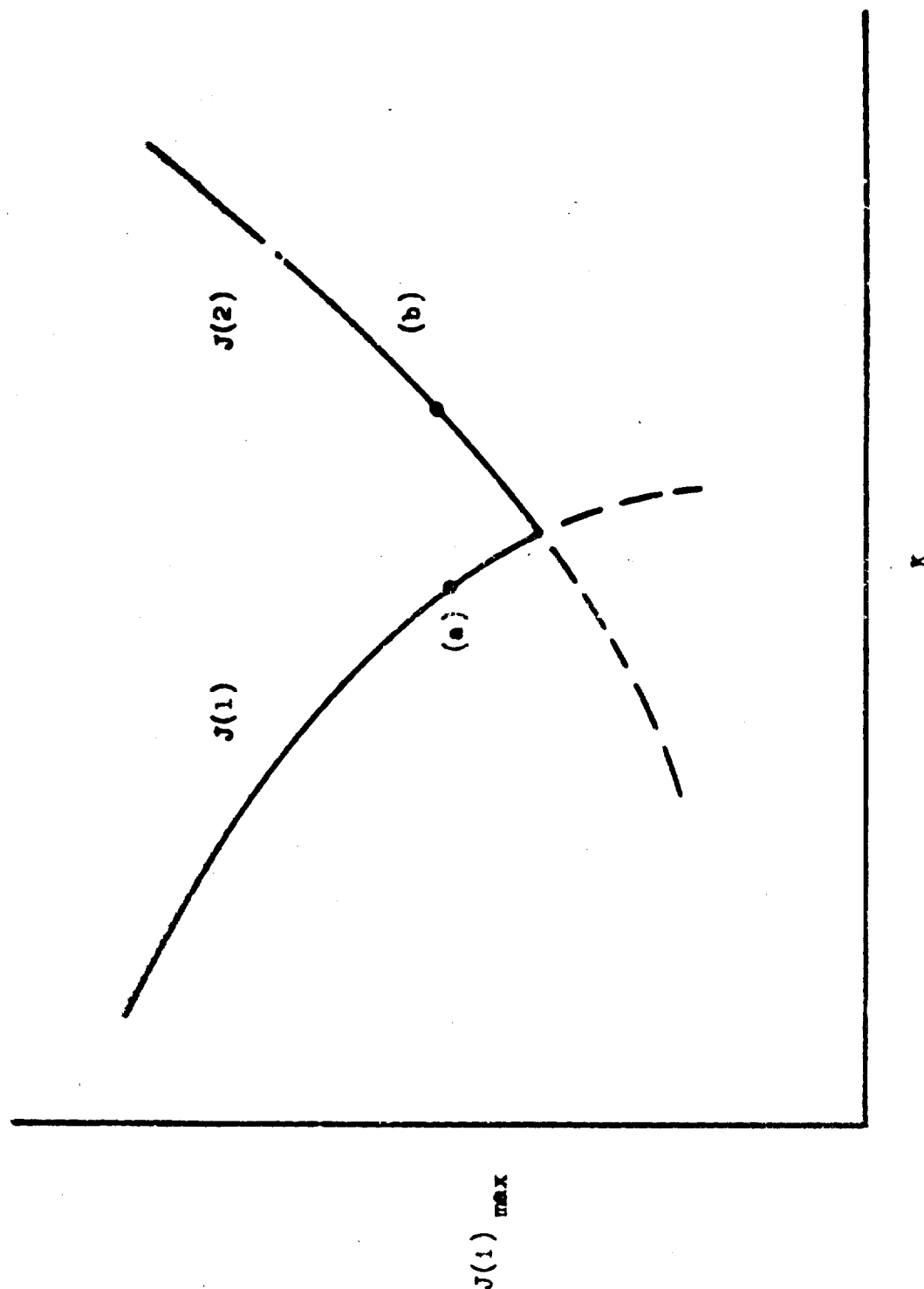
calculation for a particular starting point can be discontinued when it is obvious that a previously determined extrema is again being approached. Gelfand[27] has suggested a method which operates a little differently from this, but which has about the same end result. The presence of regions of instability within the search space may greatly complicate the steepest ascent approach. It is certainly possible that a significant fraction of the steepest ascent paths will lead into regions of instability. When this occurs, methods must be formulated to circumvent these regions.

It can therefore be concluded that steepest ascent methods can be readily applied to unimodal, one, two or possibly three-dimensional optimizations. The added complexity of the Max-Ranking Optimization Criterion, the regions of instability and the higher dimension searches which are treated here in Chapter 5 rule out the use of this method for this type of application.

4.1 Random Search Techniques

The basic concept of a random search is not new. However, improvements in digital computer speed within the past few years have now enabled random search methods to be applied to more complex problems. Other people have contributed to this field[28], but probably the most complete treatment is that given by Karnopp[29, 30]. Karnopp suggests a number of ways of employing random search. Only three general methods are discussed here.

The various types of random search methods can best be explained through the use of a simple example. Consider the unimodal function, $F(K)$, shown in Figure 9. It is required that we obtain the minimum of the function between the limits of K_a and K_d.

The first approach to the problem might be to try a purely random search. This involves making completely random choices of K between $K = K_a$ and $K = K_d$. The search probability density function which describes the selection process is,

$$p_s(K) = \frac{1}{K_d - K_a} \qquad \text{for } K_d \geq K \geq K_a \qquad (14)$$

and

$$p_s(K) = 0, \text{ elsewhere.}$$

It is easy to show that the probability of choosing a number within $\pm \Delta K_c$ units of the actual minimum at K_c is:

$$p(K_c + \Delta K_c \geq K \geq K_c - \Delta K_c) = \frac{2 \Delta K_c}{K_d - K_a} . \qquad (15)$$

As might be expected, this is one of the least efficient random search procedures.

A second approach, and one suggested by Karnopp, is to use a purely random search for only the first phase of the search. At the end of this phase, the value of K which has given the lowest value of $F(K)$ is recorded. This best previous value of K is denoted as K_b.

Figure 9

Example Function for the Demonstrated Random Search Methods
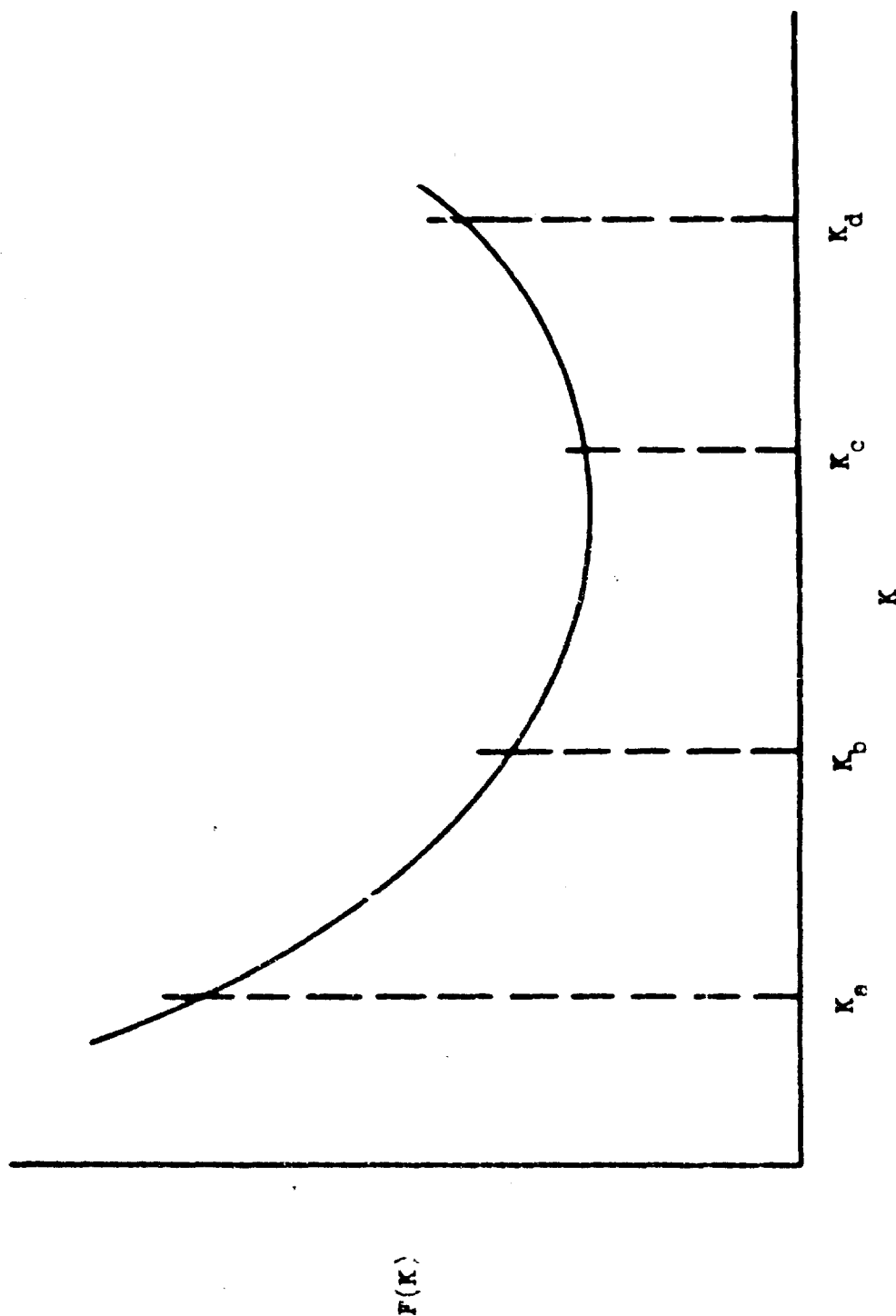
The parameter K_b takes on the new value of K each time a choice is made which produces a value of $F(K)$ lower than the previous value, $F(K_b)$. The search probability density function for the second phase is taken as,

$$p_s(K) = \frac{1}{2\Delta K_b} \qquad \text{for } K_b + \Delta K_b \geq K \geq K_b - \Delta K_b \qquad (16)$$

and

$$p_s(K) = 0, \text{ elsewhere.}$$

This provides a uniform probability, local random search which is centered around the best previous value of $K(K = K_b)$. If ΔK_b is large, this phase will behave much like the purely random search of the first phase. If ΔK_b is small, in particular, if

$$\Delta K_b \leq |K_c - K_b| \quad , \qquad (17)$$

the probability that the next choice is an improvement over the best previous value is 0.5. Since we have no prior knowledge as to the location of K_c, it is impossible to determine ΔK_b beforehand. The above relationship indicates that the smaller the value of ΔK_b, the higher the probability that the next choice will be an improvement. On the other hand, the expected change in K, $E(|\Delta K|)$, will decrease with decreasing ΔK_b,

$$E(|\Delta K|) = \frac{\Delta K_b}{2} \qquad (18)$$

- 45 -

Thus we see that small values of ΔK_b may give a higher probability of improvement, but the improvements obtained will be smaller. Karnopp[30] suggests a way to vary ΔK_b in order to improve this phase of the search procedure.

One of the underlying objectives of this work was to arrive at a method of solution which would be sufficiently general, yet be as simple as possible to implement. Therefore, an attempt was made to find a random search procedure which was more efficient than the purely random search and less complicated than the previously mentioned method. The one which has shown the most promise is the exponential random search, a method which bears some resemblance to the "creeping" random search suggested by Brooks[28].

This method employs a search probability density which is largest at $K = K_b$ and tapers off exponentially on both sides of $K = K_b$. This is expressed as,

$$ p_s(K) = \frac{1}{m(K_d - K_a)} \left[\frac{K - K_b}{K_d - K_a} \right]^{\frac{1-m}{m}} \qquad (19) $$

This search probability density is plotted in Figure 10 as a function of $K - K_b$ for several values of m. The integer m must be odd in order to produce a search probability density symmetric about $K = K_b$.

This seemingly complicated search probability density was chosen because it can be generated very easily. Most computation facilities have a simple subroutine which generates random numbers. The

- 46 -

Figure 10
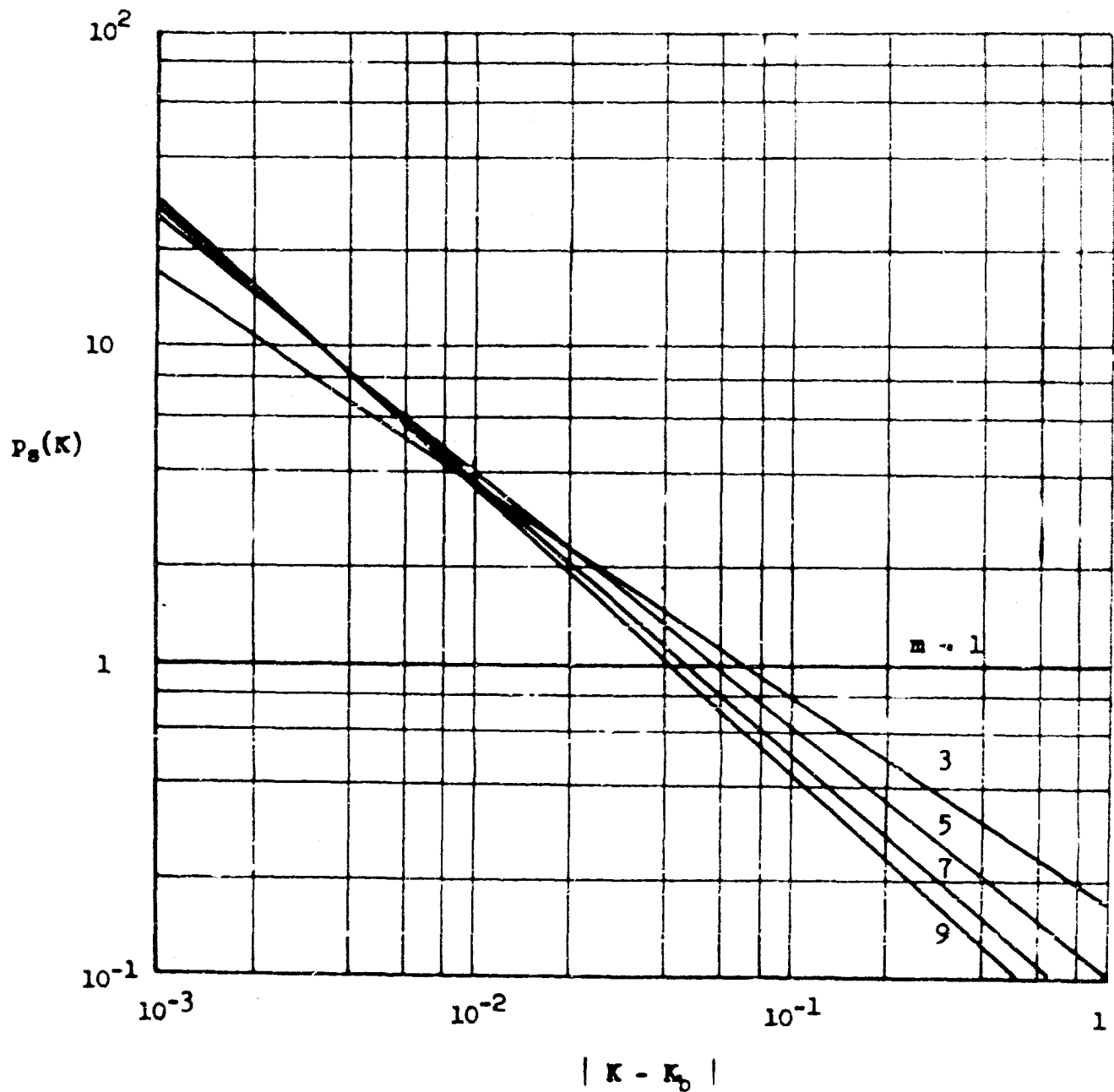
Search Probability Density

M. I. T. Computation Center subroutine is labeled RANNOF(B), where
B, for our purposes, is a dummy variable. This subroutine generates
random numbers between zero and one with uniform probability density.
With the aid of this subroutine, a new value of K can be generated
with only the following single Fortran statement:

$$AK = AKB + (AKD - AKA) * ((2.*RANNOF(B)-1.) **M),$$

where

$$AK = K,$$

$$AKA = K_a,$$

$$AKB = K_b,$$

$$AKD = K_d, \text{ and}$$

$$M = \text{the exponent } (m).$$

The probability density is somewhat altered by the fact that values
of K greater than K_d or less than K_a are disallowed. A random search
with a search probability density of this form has several advantages.
First, a certain percentage of large steps will be taken. This is
advantageous during the initial phase of a search. It is also important
for functions which are not unimodal, since it is certainly necessary to
be able to move from the area of a relative minimum to the vicinity of
the true minimum. In fact, it is suggested that the search procedure
be set up so that any point in the search space can be reached in one
step from any other point in the search space.

A second advantage is that a reasonably large percent of the time is spent searching the area immediately adjacent to the best previous point. This is particularly useful in the final stages of the search.

Although the random search method which uses the changing probability density function shares most of these advantages, the exponential random search requires fewer decisions to be made during the operation of the digital computer program and is therefore simpler to program. For this reason, the exponential random search was selected for this study.

4.2 General Properties of the Exponential Random Search

One of the difficulties with studying or comparing search techniques is the necessity of choosing some particular function on which the search can be carried out. This can lead to many mis-conceptions, since search techniques which work efficiently with one type of function may behave poorly with others. For this reason, the reader should keep in mind the fact the results obtained are for specific types of functions. Although an effort has been made to make the function as general as possible, there is, of course, no guarantee of general applicability.

Consider the function of Figure 11, which has the following properties:

1. It is symmetrical about $K = 0$.

2. Only the portion between $K = K_a$ and $K = + K_a$
   will be considered.

- 49 -

Figure 11

One-Dimensional Sample Function
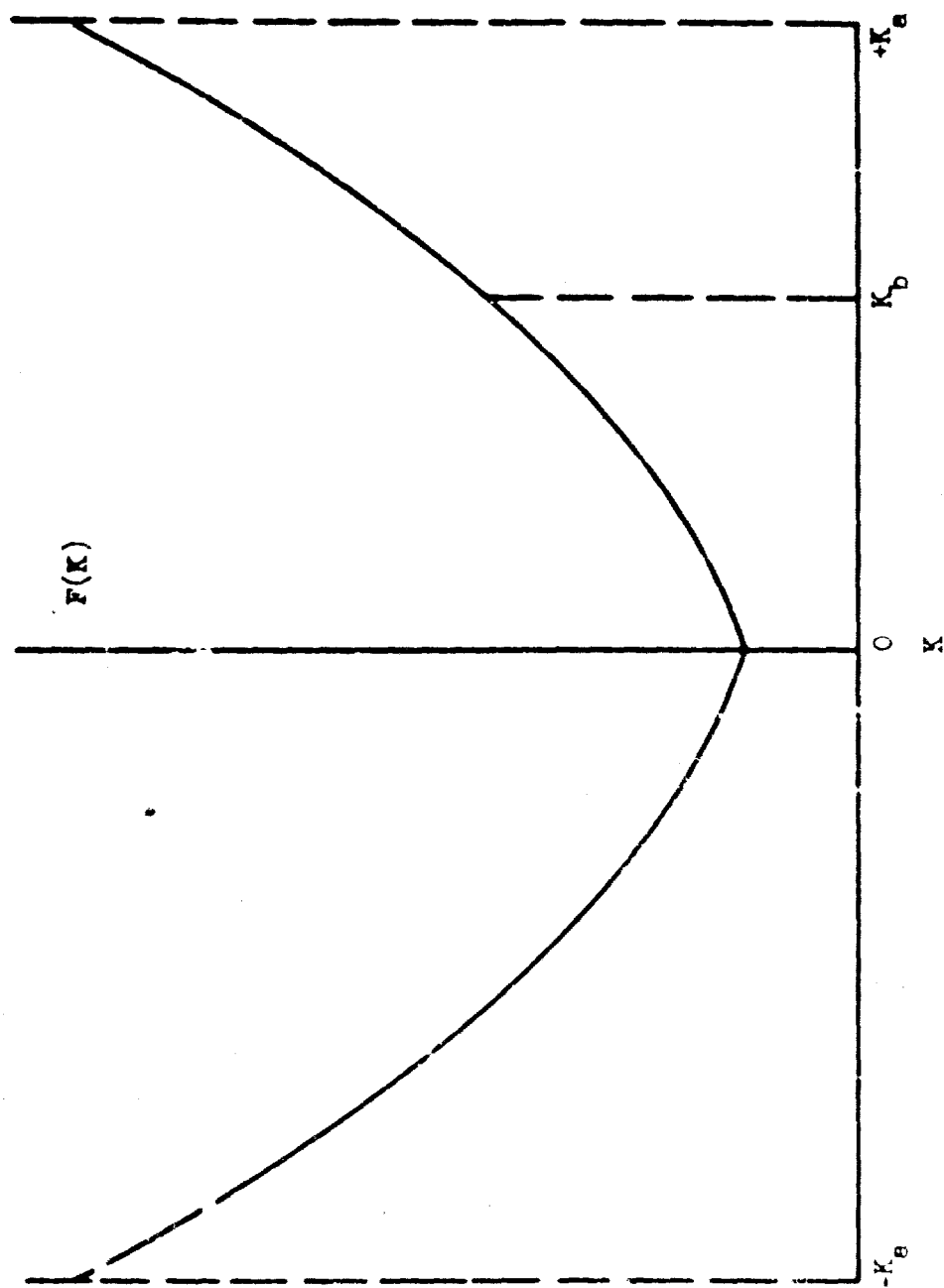
3. The function between $K = 0$ and $K = + K_a$ is monotonically increasing.

4. The function need not be continuous in slope.

The best value obtained at this point in the search is $K = K_b$. Each new value of K chosen is calculated from,

$$K = 2K_a (G_K)^m + K_b \qquad (20)$$

where m is an odd integer and G_K is a random variable having the probability density,

$$p(G_K) = 1/2 \qquad \text{for } |G_K| \leq 1 \qquad (21)$$

and

$$p(G_K) = 0 \quad \text{elsewhere.}$$

Since

$$p(K) \, dK = p(G_K) \, d \, G_K, \qquad (22)$$

the search probability density of K can be readily determined to be

$$P_s(K) = \frac{1}{2m \, K_a} \left(\frac{K - K_b}{2K_a} \right)^{\frac{1-m}{m}}, \qquad (23)$$

which is a special form of equation (19)

The probability of reaching a lower value of $F(K)$ on the next attempt is call the probability of improvement, $p_I(K_b)$. This is , equal to:

$$P_I(K_b) = \frac{\displaystyle\int_{-K_b}^{+K_b} p_e(K)\,dK}{\displaystyle\int_{-K_a}^{+K_a} p_a(K)\,dK} \tag{24}$$

Equation (23) du.s not include the effects of disallowing all values of K less than $-K_a$ and greater than $+K_a$. The denominator of equation (24), therefore, represents the fraction of all possible choices which fall within the accepted region. The inclusion of the denominator represents the fact that disallowed choices are ignored.

Equation (24) can be evaluated using equation (23), with the result,

$$P_I(K_b) = \frac{(K_b/K_a)^{1/m}}{\left(\dfrac{1 + K_b/K_a}{2}\right)^{1/m} + \left(\dfrac{1 - K_b/K_a}{2}\right)^{1/m}} \tag{25}$$

which is plotted in Figure 12 for several values of m. For comparison purposes, the value of m = 1 was also included. This represents the ·· ·. y ranium cearch. .he other extreme is represented by m = • which produces a value of $p_I(K_b)$ equal to 0.5 for all values of K_b/K_a except zero and 1. It is interesting to note the probability of improvement for various values of m when K_b/K_a = .01.

Figure 12

Probability of Improvement*

* for a one-dimensional
exponential random search
of a unimodal function

$F_I(K_b/\overline{K}_a)$

K_b / K_a
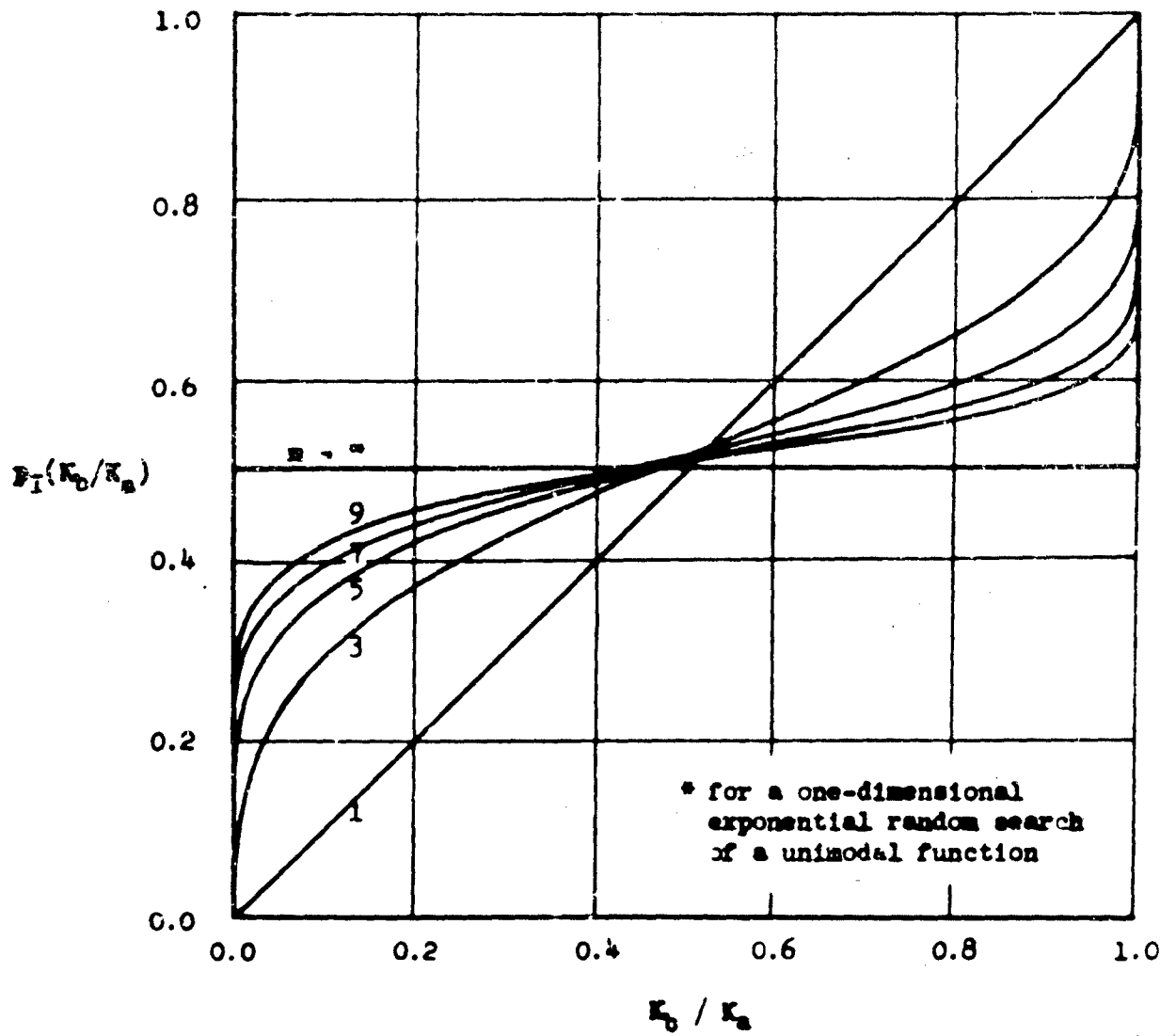
m	$p_I(K_b)$
1	0.01
3	0.136
5	0.229
7	0.286
9	0.324
∞	0.50

Table 2

Probability of Improvement at the 1% Point ($K_b/K_a = .01$)

This indicates why high values of m are better in the final stages of a search.

A second characteristic which is important is the expected step length, $E(|\Delta X|)$. The dimensionless expected step length is equal to:

$$E\left(\frac{|\Delta K|}{X_a}\right) = \frac{\int_{-K_a}^{+K_a} \frac{|K - K_b|}{K_a} p_s(K)\, dK}{\int_{-K_a}^{+K_a} p_s(K)\, dK} \tag{26}$$

This can be integrated in a fashion similar to the previous example.

$$E\left(\frac{|\Delta K|}{K_a}\right) = \frac{2}{m+1} \quad \frac{\left(\dfrac{1 - K_b/K_a}{2}\right)^{\frac{m+1}{m}} + \left(\dfrac{1 + K_b/K_a}{2}\right)^{\frac{m+1}{m}}}{\left(\dfrac{1 - K_b/K_a}{2}\right)^{\frac{1}{m}} + \left(\dfrac{1 + K_b/K_a}{2}\right)^{\frac{1}{m}}} \tag{27}$$

This relationship is plotted in Figure 13.

There is one other piece of information which can be obtained from this general unimodal function. This is the probability density of the location of the current best value (K_b) after N_c choices have been made. Since the probability density of the position of the next point to be chosen in only a function of the current location of K_b, this process is called a Markov process[31]. A random process described by $K(t)$ is a Markov process if and only if, for every finite set $t_1 < t_2 < \ldots < t_{n-1} < t_n$,

$$p(X_n, t_n | X_1, t_1; \ldots; X_{n-1}, t_{n-1}) = p(X_n, t_n | X_{n-1}, t_{n-1}) \tag{28}$$

A very convenient property of a Markov process is this fact, that the probability density at any state is only dependent upon the probability density at the previous state and the transition probability density. Therefore, given the initial starting point and the probability density which describes the incrementing or stepping process (Figure 10), it is theoretically possible to determine the probability density of K_b for each step in the search process. In practice, it is not possible

- 55 -

Figure 13

Expected Step Length*

* ... one-dimensional
  ... of a
  ... function

$n = 1$

3

5

7

$\varepsilon\left(\frac{|\Delta K|}{K_a}\right)$

K_b/K_a
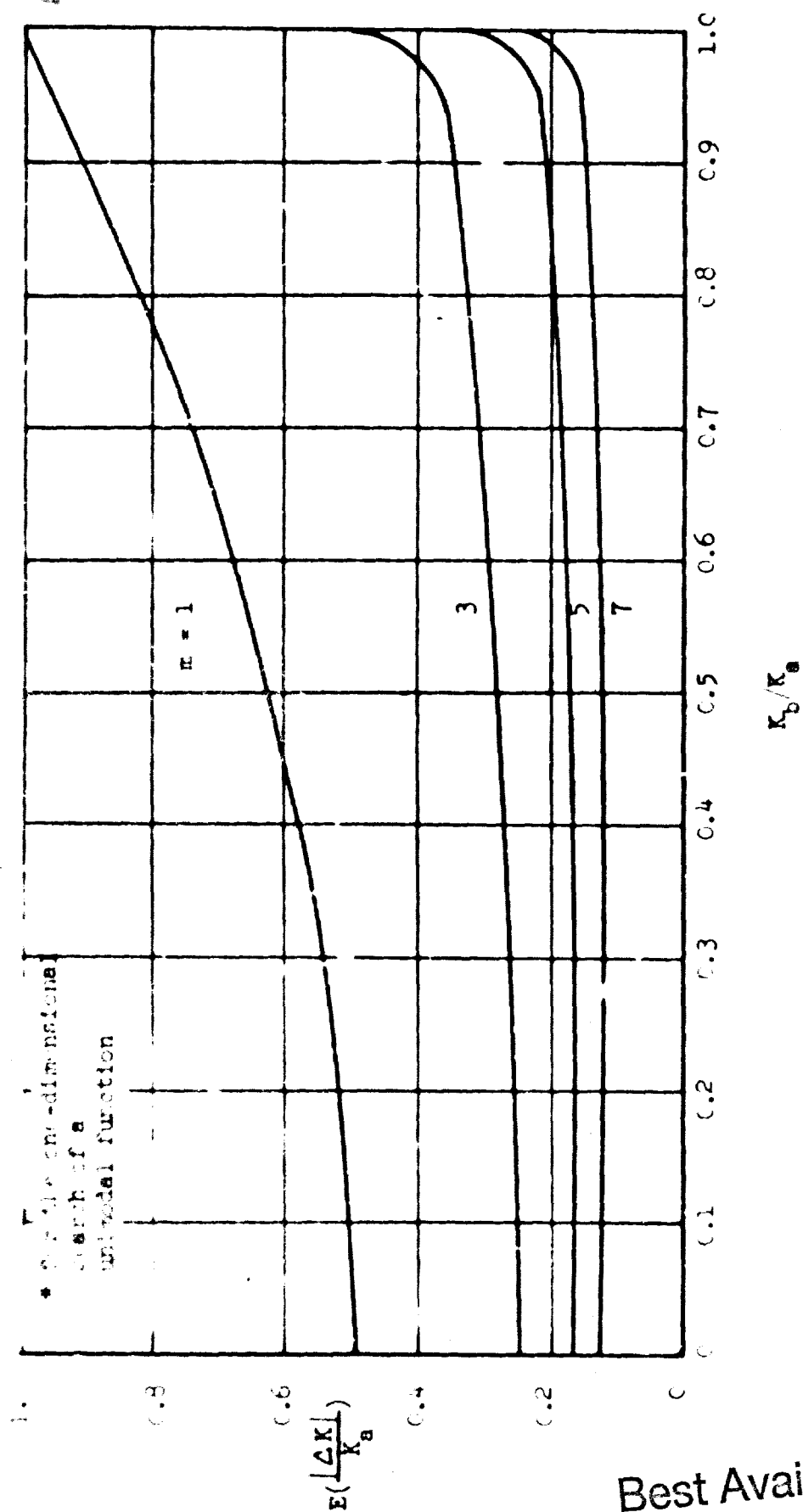
to do this directly because the integrals involved very rapidly grow too complex. Therefore, it is first necessary to transform the several continuous probability density function into discrete functions. The probability of the ith discrete value of K_b after N_c choices ($p(K_b, N_c)$), given $p(K_b, N_c - 1)$ is obtained as follows:

1. Multiply the probabilities at step $N_c - 1$ by their corresponding probability of improvement, $p_I(K_b)$, a discrete-variable version of Figure 11. This gives the portion for which there will be an improvement.

2. This portion is then distributed over the allowable regions according to the exponential search probability distribution. (By "allowable" we mean that portion of the function $F(K)$ which is less than $F(K_b)$).

3. Repeat step 2 for all values of $K_b(i)_{N_c-1}$ and at each point $K_b(i)_{N_c}$ sum the contributions from all such points.

4. Multiply the probabilities at step N_c-1 by $(1 - p_I(K_b))$. This gives the portion for which there will be no improvement.

5. Add this static portion to the changing portion obtained in step 3.

- 57 -

This process was programmed for a digital computer and results were obtained for two values of the exponent m (m = 3 and n = 9). The results are shown in Figures 14, 15 and 16.

The results demonstrate the general characteristics which can be predicted qualitatively on the basis of Figures 12 and 13, that is, low values of m produce more rapid initial convergence, but slower final convergence than higher values.

It should be remembered here that these results are for unimodal functions. The problems of multimodal functions are more complicated by an order of magnitude. It is impossible to find a value of the exponent m which is optimum for all multimodal systems and it is ridiculous to obtain the optimum value of m for one particular function. Some general recommendations can be made, however. These are presented in Section 4.4

An exponential random search for a minimum might be called an "Inebriated Random Walk". In this case, the often-discussed alcoholic still cannot control his direction or step length, but he at least realizes that he is in no shape to walk uphill.

4.3 Multidimensional Random Search

One of the basic attributes of random search methods is the fact that the programming of a general n-dimensional search is no more complex than the programming of the two-dimensional search. The following five Fortran statements are sufficient to select the new trial point for the general n-dimensional case:

- 58 -

Figure 14

Probability Distribution of K

after N_c Choices

$N_c = 1$ m = 3, 1% increments

$N_c = 10$

$N_c = 25$

$N_c = 50$

K/K_e
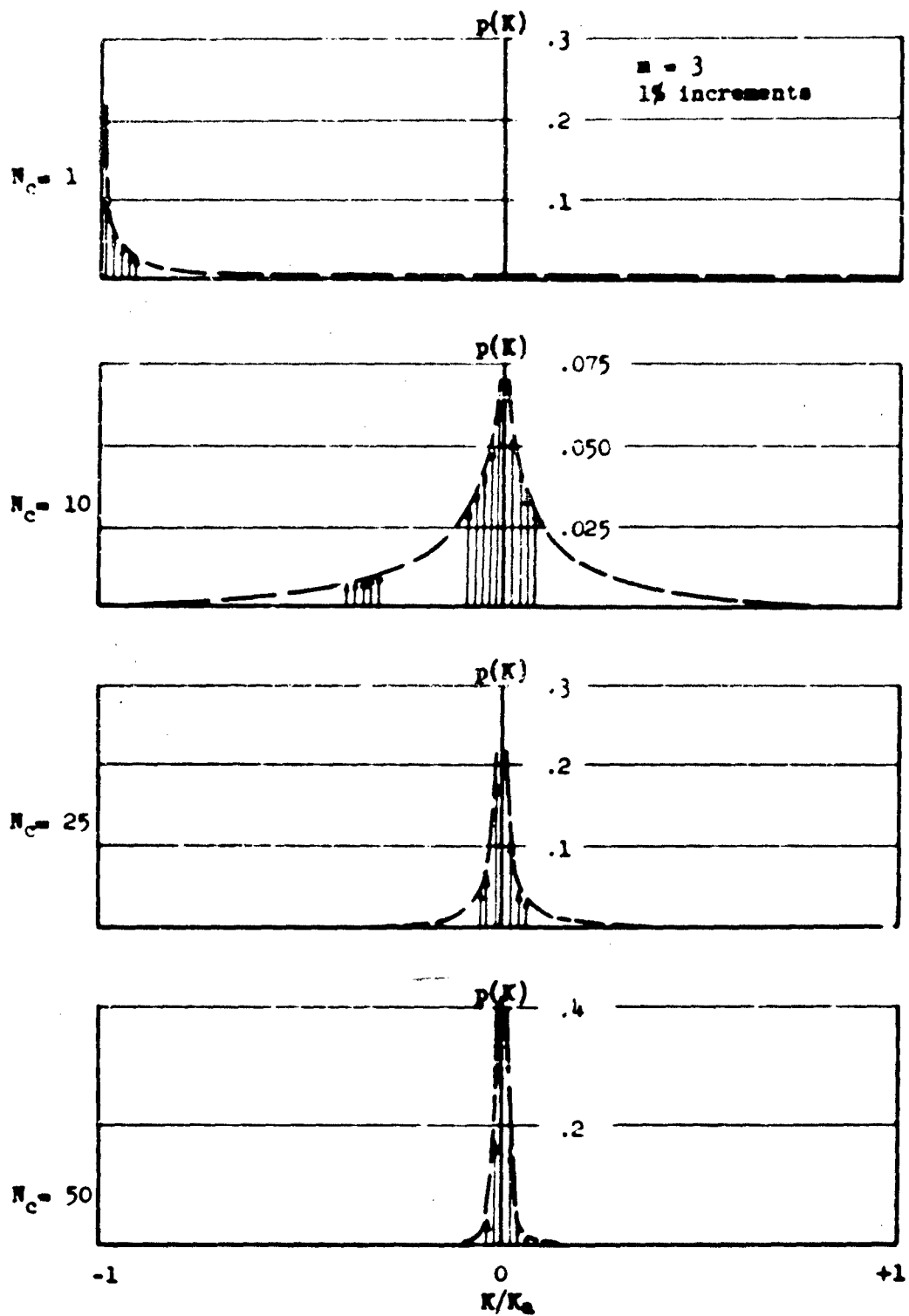
Figure 15

Probability Distribution of K

after N_c Choices

p(K)

$N_c = 1$

$n = 9$
1% increments

.6
.4
.2

p(K)

$N_c = 10$

.03
.02
.01

p(K)

$N_c = 25$

.15
.10
.05

p(K)

$N_c = 50$

.3
.2
.1

-1   0   +1

K/K_a
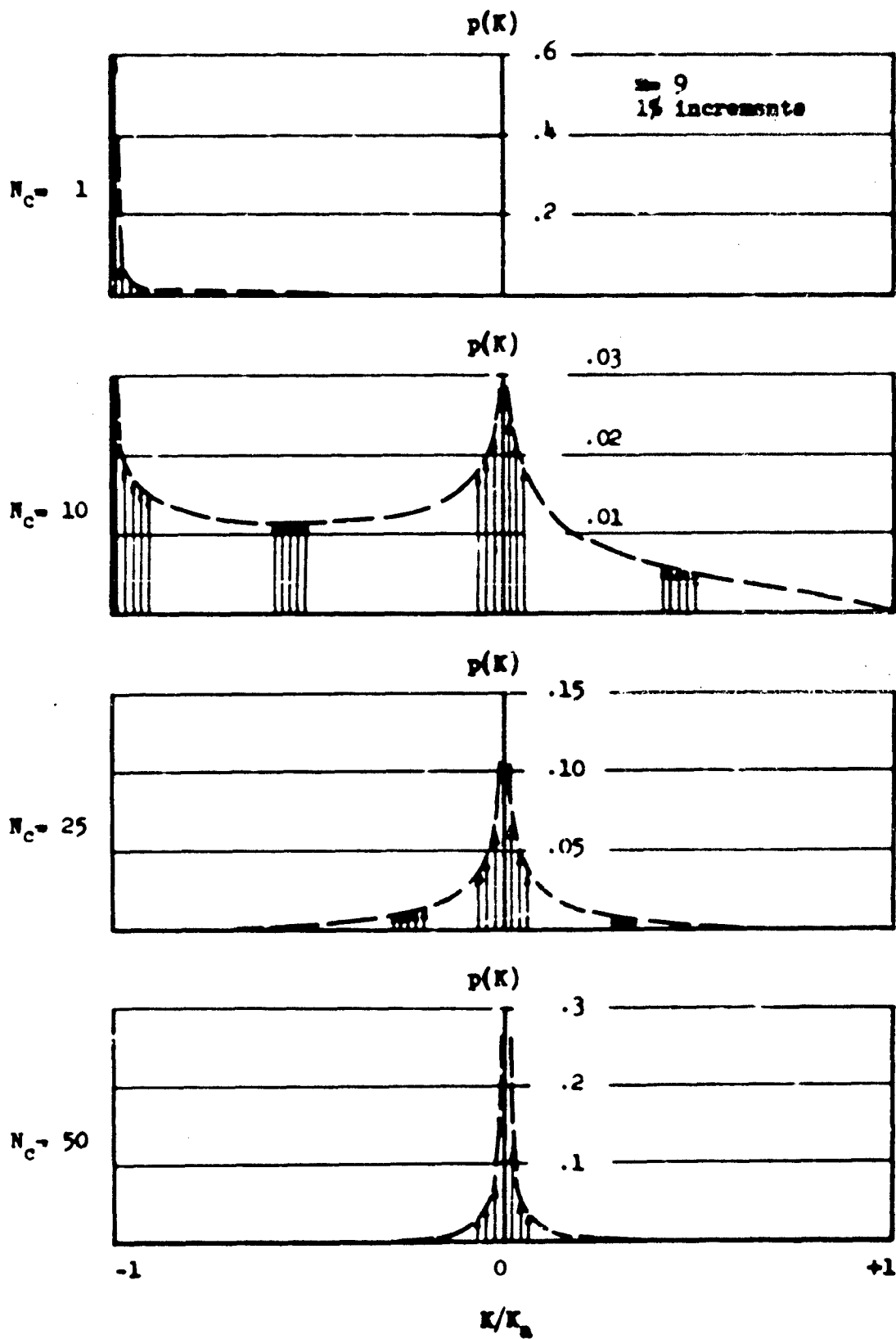
Figure 16

Relative Speed of Convergence

of Two Exponential Random Searches

$p(|K_b/K_a| \leq .01)$

m = 3

m = 9

Number of Choices, N_c
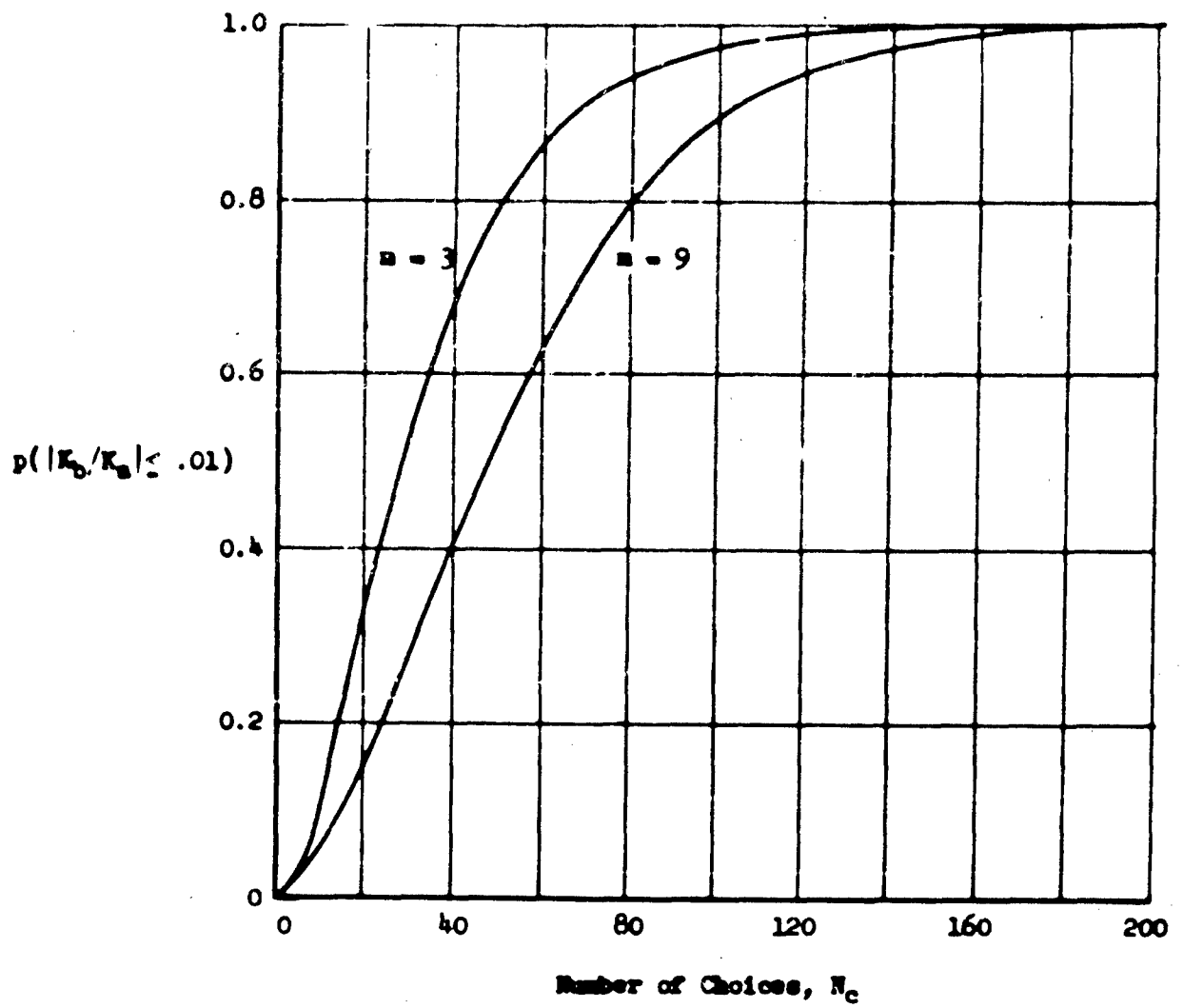
```
1.  DØ 5  I = 1, N

2.  AK(I) = AKB(I)+(AKD(I)-AKA(I))*((2.*RANNØF(B)-1.)**M)

3.  IF(AK(I)-AKA(I)) 2, 4, 4

4.  IF(AKD(I)-AK(I)) 2, 5, 5

5.  CØNTINUE
```

The subscript (I) refers to the Ith dimension $(I = 1$ to $n)$. The remaining terms are as previously defined.

Although the simplicity is maintained for the higher dimensional search, the search efficiency can be expected to deteriorate. Figure 17 shows one quadrant of the two-dimensional search probability density. It is obvious from this figure that this type of search profile provides a more efficient search along the principal axes than in the diagonal directions. During the course of this work, both spherical and ellipsoidal profiles were tried in an attempt to produce a more rotationally-balanced search probability density. These methods worked satisfactorily for two dimensions, but the increase in complexity for higher dimensions was not worth the somewhat doubtful gain in search efficiency.

The worst multidimensional search for the exponential probability density is one which would continue to move along a diagonal. The best case is one which would continue to move along the principal axes. Thus, these two extremes must certainly bracket all possible cases.

Figure 17

Two Dimensional

Search Probability Density

$$a = \left[16\pi^2 \; K_{1a} K_{2a} \; p(K_1, K_2) \right]^{\frac{1-m}{m}}$$
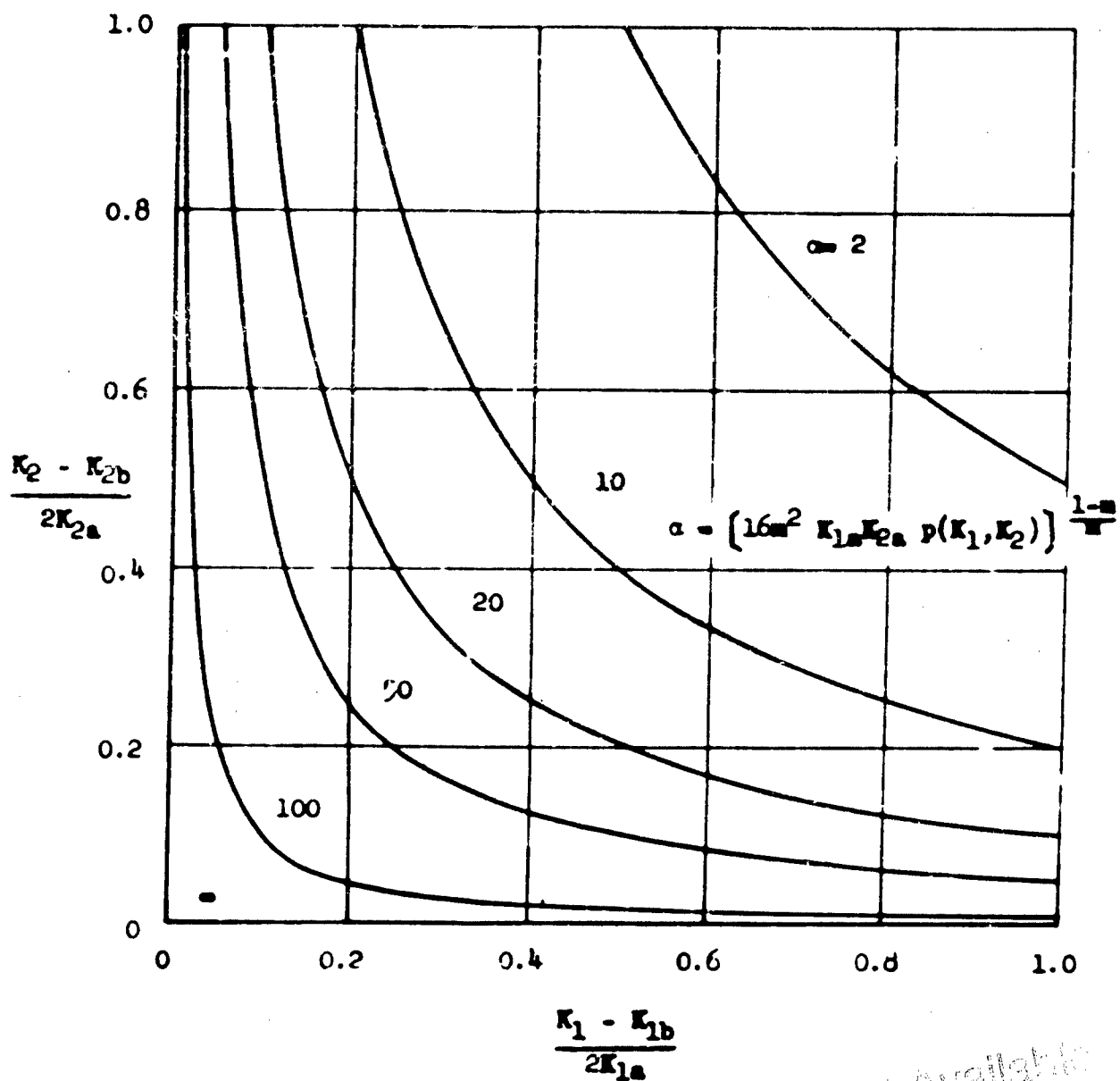
The probability of improvement for the diagonal search is plotted in Figures 18 and 19 for several values of n and two values of m. The one-dimensional search (n = 1) is also plotted since it represents the maximum search efficiency in each case. From this it is obvious that the number of trials must necessarily increase with the dimensionality of the problem. An empirical formula has been found to be helpful for determining how many test points are necessary to give a reasonable confidence in the search result:

$$(N_c)_n \approx (N_c)_1 (2)^{n-1} \tag{29}$$

This states that the number of choices for an n-dimensional search is roughly equal to the number required for a comparable one-dimensional search times 2 to the (n - 1) power. The number of choices (N_c) refers to the number of choices which do not fall into a region of instability. This empirical relationship is based on a large number of one through five-dimensional studies. The few runs which were made at higher values of n (up to and including 9) tend to show that this is too pessimistic. This is certainly one area in which more work is needed.

4.4 General Recommendations for Applying Random Search Techniques

Very little information is available on the application of random search methods to practical engineering problems. It is the purpose of this section to list some of the practices which evolved during this study. In addition to the three applications described in

Figure 18

Probability of Improvement*
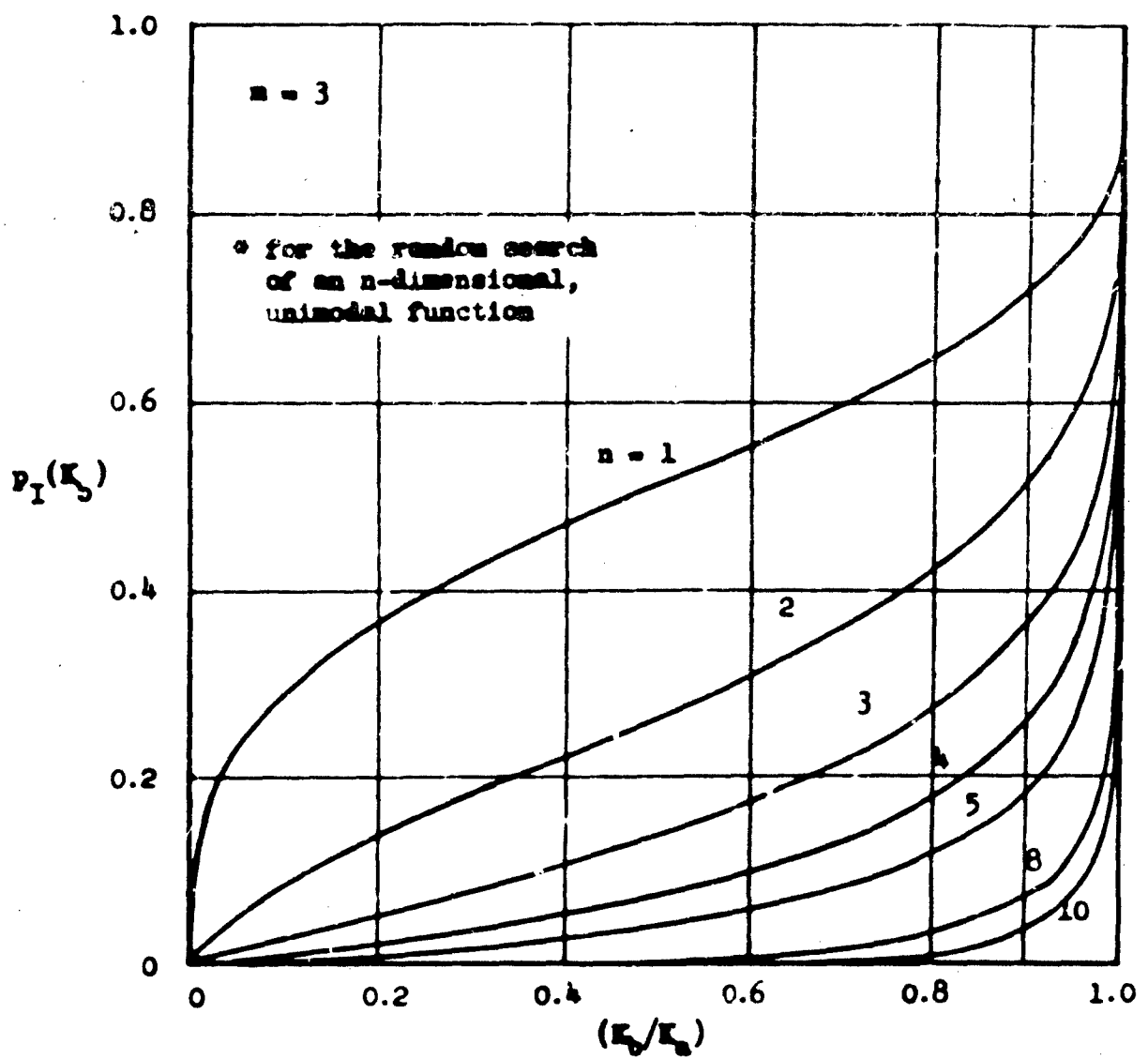
Figure 19

Probability of Improvement*

Figure shows a graph with vertical axis $P_I(K_b)$ ranging from 0 to 1.0 and horizontal axis (K_b/K_a) ranging from 0 to 1.0.

Annotations within graph:
- $m = 9$
- * for the random search of an n-dimensional, unimodal function
- $n = 1$
- 2
- 3
- 4
- 5
- 8
- 10
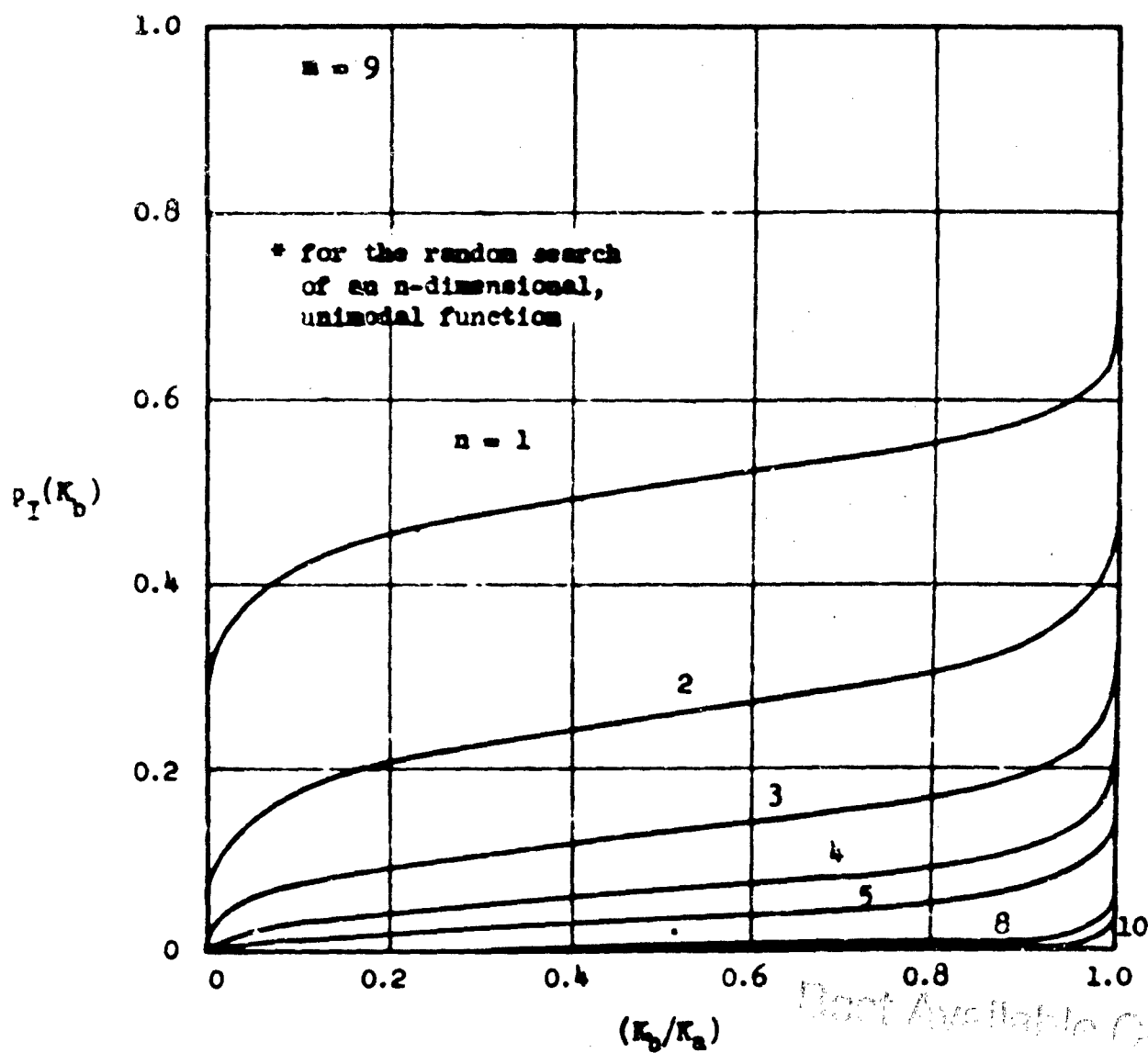
Chapter 5, a number of multidimensional, unimodal functions were studied. This was done in an attempt to become more familiar with the behavior of random search techniques for known functions, so that an easier transition could be made to the more complex engineering problems. This turned out to be time well-spent. Many of the following recommendations are based upon information learned during this phase of the study.

1. For low dimensional searches (n = 1 through 3), an exponent (m) of 3 or 5 will provide sufficiently rapid convergence for most practical problems. If accuracies better than $\pm$ 1% are required, the final portion of the search can be conducted using an exponent of 7 or 9.

2. Convergence to within $\pm$ 1% of the actual extremum can be expected within 100 choices $((N_c)_1 = 100)$ for a one-dimensional search of a system which is not interrupted by unstable regions for an exponent (n) equal to 3.

3. More choices are required for systems which have a higher percentage of the search space taken up by unstable regions. Evidentally, the disturbance of the normally smooth search "flow" is of more importance than the benefit obtained from the reduction of the search space.

4. There are at least two equally acceptable methods for terminating the search:

  a. Stop after a given number of choices (N_c). (This does not include any choices which fall in unstable regions.)

  b. Stop after a given number of stable choices fail to produce an improvement of a certain percent (for example, .01%).

5. For complex, multimodal, multidimensional problems, it is best to run at least two short, less exact $(m = 3)$ searches starting from different positions. If these tend to converge in one location, there is a high probability that the true extrema is being approached. A higher exponent search $(m = 7$ or $9)$ can then be initiated at the end position of one of the previous searches. If two or more locations result from the initial searches, each of these may have to be searched more exactly to determine the true minimum. This is usually a more satisfactory procedure than running one, long, exact search.

6. The search efficiency can usually be improved by running a purely random search $(m = 1)$ for the first 10 or 20% of the total search, an exponential search with $m = 3$ for the next 40 to 60% of the search, and

an exponential search with m = 7 or 9 for the
remainder of the search.

7. For complex systems containing several regions
of instability, it may be desirable to determine the
shape of the function in the vicinity of the
extremum. This will indicate how sensitive the
function is to small changes in each of the con-
trol variables as well as pointing out possible
regions of instability which may be immediately
adjacent to the extremum.

5.0 Applications

In this chapter, we will attempt to show how the theories of the previous three chapters can be put to use to solve practical engineering problems. Three problems were chosen as examples.

The first (section 5.1) deals with the positioning control of a typical second order system (mass-spring-dashpot). The random disturbance has a damped-exponential autocorrelation function and is produced by a first-order, low-pass filter which is subjected to zero-mean, Gaussian white noise. Acceleration, velocity, and displace- m t feedback control are employed and the maximum control force is limited to $\overset{+}{-} F_{max}$.

The second and third problems both deal with a submarine which is running at periscope depth in a random sea. In the second example, the optimal roll control of the submarine is determined. The power spectral density of the roll forcing function is calculated from the Neumann wave height spectrum[32, 33, 34] and from submarine geometry. A control with five variable parameters is designed.

In the third problem, the coupled pitch-heave power spectra are determined in a fashion similar to that of the preceeding problem. The nine, variable parameters of the control system are specified so as to optimize a given system performance criterion.

The first example was chosen to compare the Max-Ranking, statistically-equivalent linearization approach to another method which did not use a criterion that minimizes some function of the error. The

purpose of this comparison was to show that the Max-Ranking Criterion
can accomplish the end goals of other criteria which necessitate more
complex mathematical analyses. The second two examples were chosen for
two reasons. First, to demonstrate how the use of the techniques
developed in this work simplify the optimization of complex systems; and
secondly, t: demonstrate that these techniques can produce practical
results for real problems.

5.1 Three-Parameter Positioning Control

The system of Figure 20 is a positioning control which has
acceleration, velocity, and displacement feedback. The input distur-
bance is formed by passing zero-mean, Gaussian, white noise through a
first order lag, or low pass filter.

This was the system us·d by Broniwitz[35] to compare the
saturating control system of Newton[6] to the "bang-bang" control
systems developed by Bass[36] and Davis[37]. (Bass and Davis both
used Lyapunov's Second Method[13, 38] to ontain the switching functions
which produce a minimum mean square error.) Newton developed a method
for minimizing the mean square erroi of a system subject to constraints
on the mean square values of one or more of the system variables (see
section 2.1.3). A solution was obtained by using Lagrange Multiplier
and Calculus of Variations techniques. Broniwitz applied Newton's
method to a second order system and obtained the control shown in
Figure 20 with the control parameters K_1', K_2' and K_3'. For a damping

Best Available

Figure 2(

Three Parameter Positioning Control
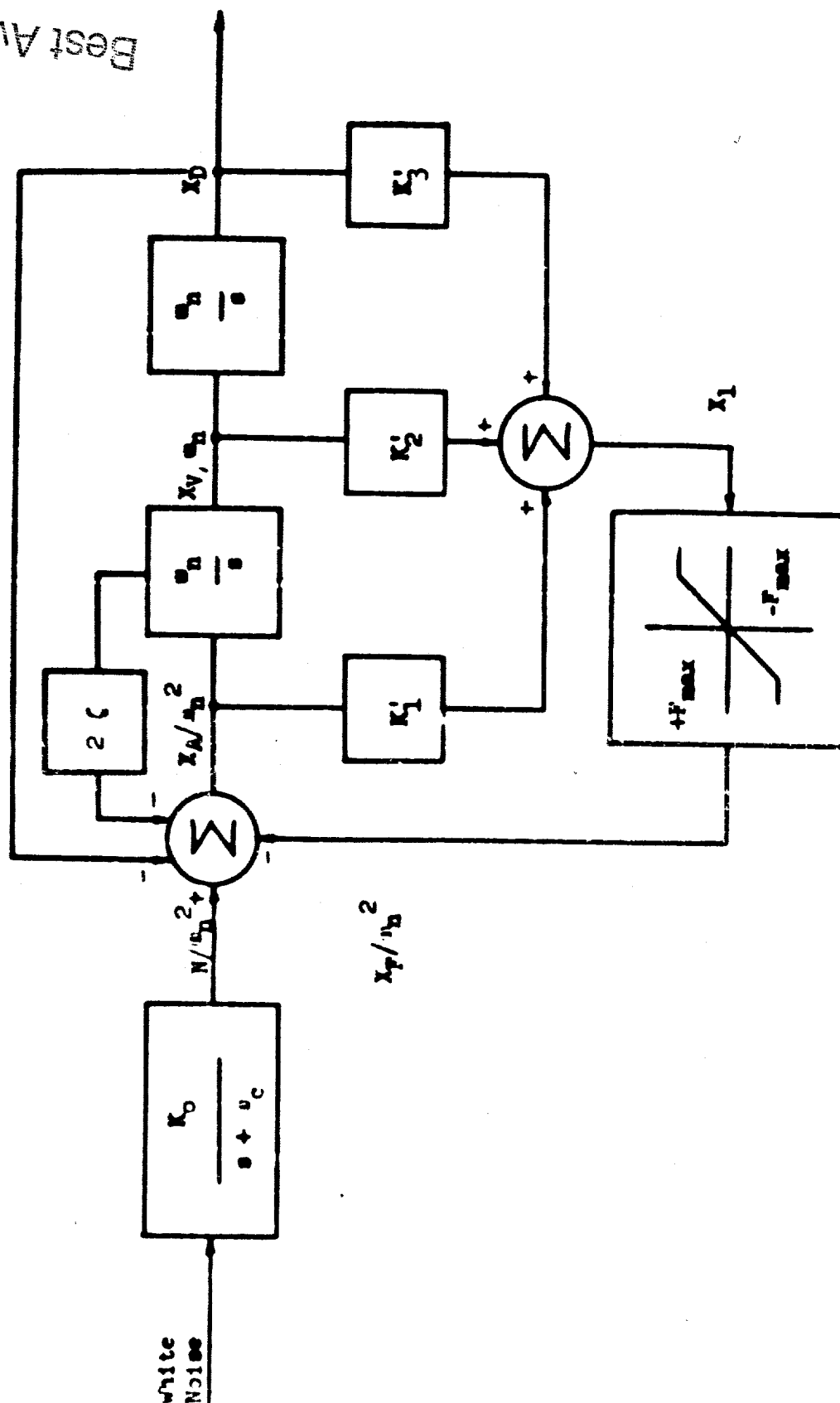
ratio (ζ) equal to 0.2, $\omega_n = \omega_c = 1000$ rad/sec., and the Lagrange Multiplier which produced the best results ($\rho = 0.1$), the control parameters for Newton's control were:

$$K_1' = 1.71,$$

$$K_2' = 5.53,$$

$$K_3' = 8.00.$$

Broniwitz then ran this system on an analog computer. Unfortunately, he only determined the two ratios: $(\sigma_{X_D}/\sigma_{X_{Do}})$ and (F_{max}/σ_{X_1}) at three values of (F_{max}/σ_n). The terms in these ratios are defined as:

σ_{X_D} — the r.m.s. value of X_D,

$\sigma_{X_{Do}}$ — the r.m.s. value of X_D when F_{max} is zero,

F_{max} — the maximum allowable value of the control force (X_F/ω_n^2),

σ_{X_1} — the r.m.s. value of X_1,

σ_n — the r.m.s. value of (N/ω_n^2).

Thus it was necessary to determine values for σ_{X_A}, $\sigma_{X_{Do}}$, and σ_{X_F} before comparing the two criteria.

it was decided to make the calculations on the digital computer using the equivalent linearization technique. This also allows us to compare some of the results from the equivalent linearization with the analog results of Broniwitz. Digital computer calculations were made for several values of K_{EQ}. The results are shown in Figures 21 and 22.

The agreement here is quite good when one considers that one of the three feedback loops passing through the nonlinearity is an algebraic loop with a loop gain of $-1.71 K_{EQ}$. This causes the signal at the input to the nonlinearity to be appreciably distorted for all values of F_{max}/c_{X_1} less than 2 or 3, which is contradictory to the basic assumption that X_1 is Gaussian. The difference between Booton's method and the first method of Kazakov is small, although Booton's method is better in this case.

Three system variables are used in the optimization criterion. These are the mean square acceleration $(\sigma_{X_A}^2)$, the mean square displacement $(\sigma_{X_D}^2)$ and the mean square control force $(\sigma_{X_F}^2)$. Results were obtained for three different Max-Ranking Arrays. They are shown in Table 3.

Array number 1 was chosen to illustrate equal emphasis between the three system variables. Arrays 2 and 3 should accomplish the basic objective of Newton's method, that is to minimize the mean square displacement while fixing an upper limit on the mean square value of the control force. The two limits were taken from the results

- 74 -

Figure 21

Comparison of Digital Results

with the Analog Simulation
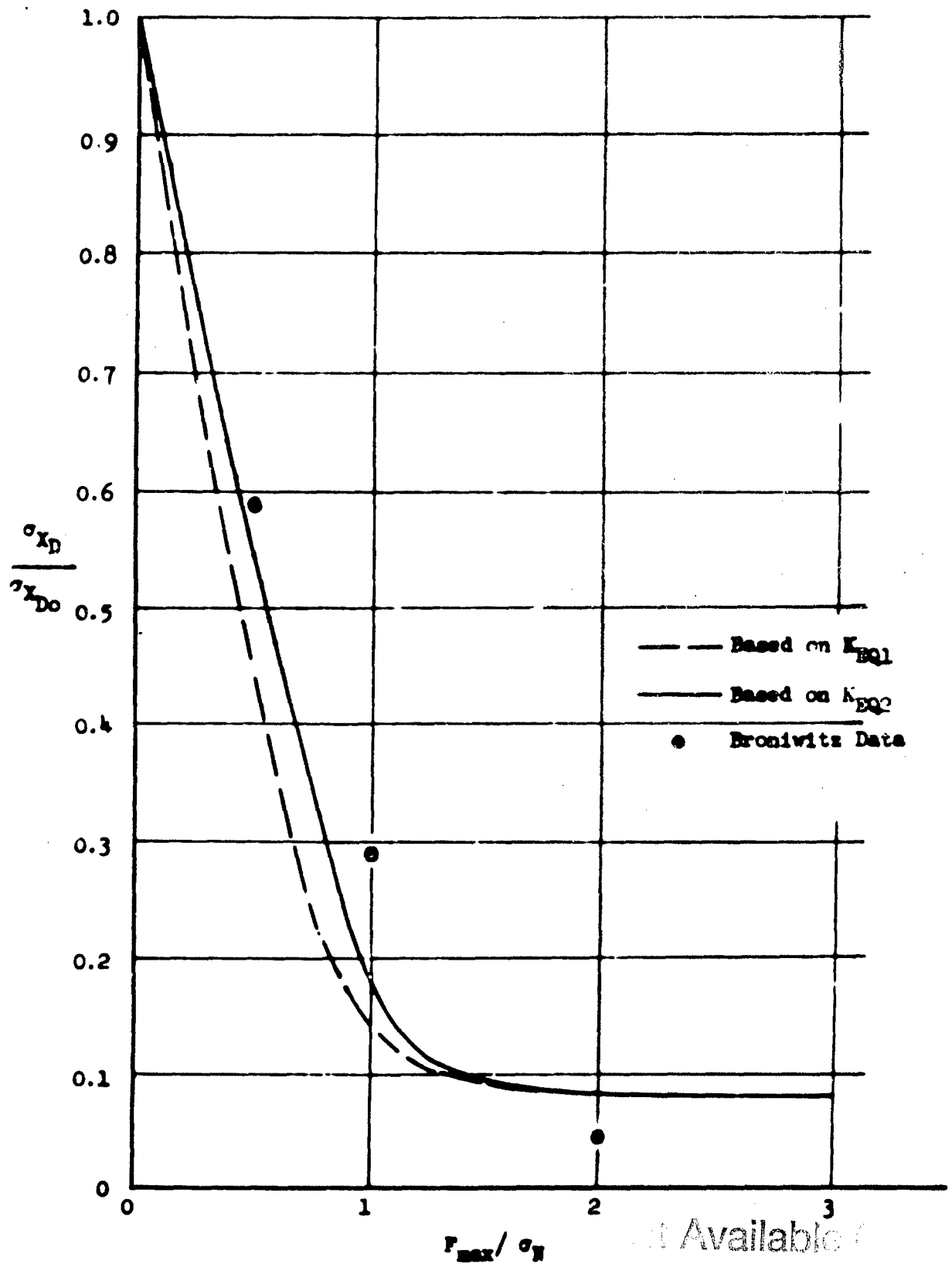
Figure 22

Comparison of Digital Results

with the Analog Simulation

Based on K_{EQ1}

Based on K_{EQ2}

Bronfwitz Data

P_{max}/σ_N^2

$\dfrac{P_{max}}{\sigma_{X_1}}$

- 2 -
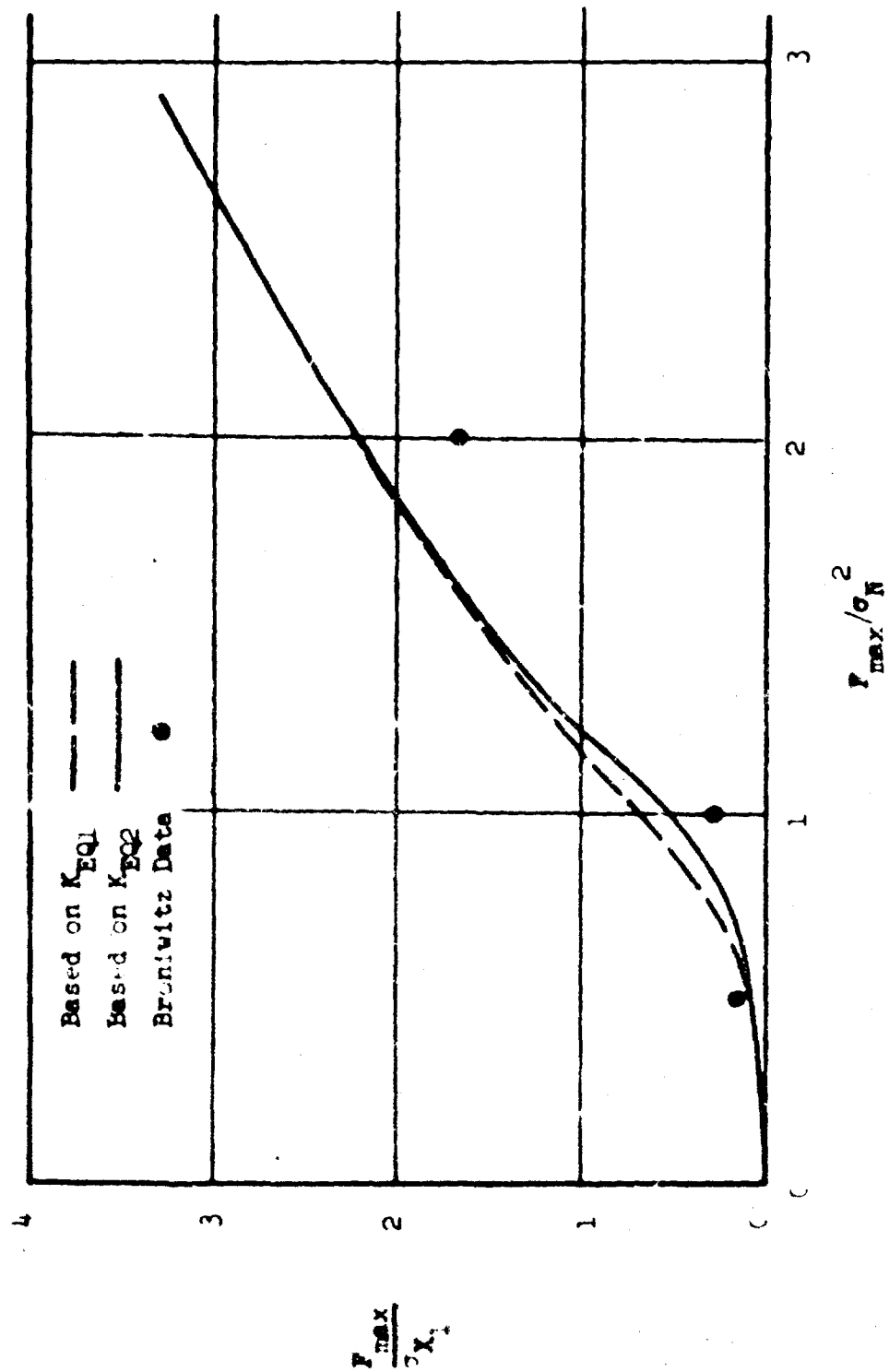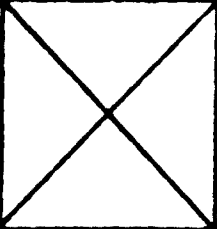
Array Number	$\sigma_{X_A}^2$	$\sigma_{X_D}^2$	$\sigma_{X_F}^2$
1	0 50	0 50	0 50
2		0 0 10 50	0 3.86 ⁻.86 50
3		0 0 10 50	0 3.17 3.17 50

Table 3

Max-Ranking Arrays for

Example 1

obtained using the control parameters determined by Broniwitz.

A computer program was written to optimize this system for each of the three Max-Ranking Arrays. A linearized approach was taken, with the direct evaluation of the third order equations as presented in Newton, Gould and Kaiser[20]. A three-dimensional random search was used to determine the three control parameters. This digital computer program is included in Appendix A-2 to show the simplicity and brevity of this approach. A sufficient number of comment cards (distinguished by the C in the first column) are included to explain the logic and mnemonic coding.

The results of these optimizations are shown in Table 4. Array 1, which specified that all three variables were of equal importance, produced a final result with $\sigma_{x_A}^2 = \sigma_{x_D}^2 = \sigma_{x_F}^2 = 1.35$. Array 2 and 3 essentially duplicated the results obtained using Newton's Method, thus accomplishing the major purpose of this example. This shows that the simple approach offered by the Max-Ranking Criterion can accomplish the same end goals as other, more complicated methods. At the same time, it never forces the designer to choose values for weighting functions or Lagrange Multipliers which have little or no physical relationship to the actual problem.

5.2 Control of Ship Motions

Before going on to the two specific problems to be discussed in this section, it is necessary to first provide some of the background information in this field. A large number of papers have been written

Control	F/σ_N	$K'_1 \, K_{EQ}$	$K'_2 \, K_{EQ}$	$K'_3 \, K_{EQ}$	K_{EQ}*	$\sigma^2_{X_A}$	$\sigma^2_{X_D}$	$\sigma^2_{X_F}$	$\sigma^2_{Y_v}$
Max-Ranking Array 1	5.0	.73	.98	.73	.999	1.35	1.35	1.35	.75
Newton's	1.67	.6	2.77	4.00	.500	.694	.158	3.17	.153
Max-Ranking Array 2		1.0	4.04	3.775		.541	.154	3.17	.129
Newton's	5.0	1.71	5.53	8.00	.999	.263	.046	3.86	.048
Max-Ranking Array 3		1.2	5.73	7.78		.277	.046	3.86	.047

*K_{EQ} was determined by Booton's Method

Table 4

Comparison of Max-Ranking Method with Newton's Method

on the response and control of a ship or submarine in the sea. Some
of the more important works are listed in references 39 to 44.

Until the last decade, almost all of the work was done based
on a "regular " sea, that is one which is perfectly sinusoidal.
Recently, people became interested in determining the statistical pro-
perties of a random sea. Many wave height measurements were recorded,
but no general correlation was obtained until Neumann arrived at a
formula[32 - 34] which describes the power spectra of a fully-arisen sea
as a function of the wind velocity (V_w) and the ship's heading (ψ_w).

$$S^2(\omega, \psi_w) = \frac{c_0}{\omega^6} e^{-(2g^2/\omega^2 V_w^2)} \cos^2 \psi_w \qquad (30)$$

The parameter, ω, is the wave frequency and C_0 is a constant equal
to 32.9 ft.^{2}/sec.^{5}. This represented a gigantic step forward. However,
the problem of determining the transfer function between the sea spectrum
and the force and moment input to the ship still remains to be solved.
Thus far, this can only be done for mathematically simple shapes such
as ellipsoids and spheroids.

Havelock[45] has developed the equations to determine the
heaving force and pitching moment of a prolate spheroid submerged in
a regular sea. Assuming that the shape of a submarine can be approx-
imated by a prolate spheroid, it is now possible to determine the power
spectral density of the pitching and heaving forcing function. The
process is shown in block diagram form in Figure 23.

Figure 23

Heave and Pitch Disturbance Model
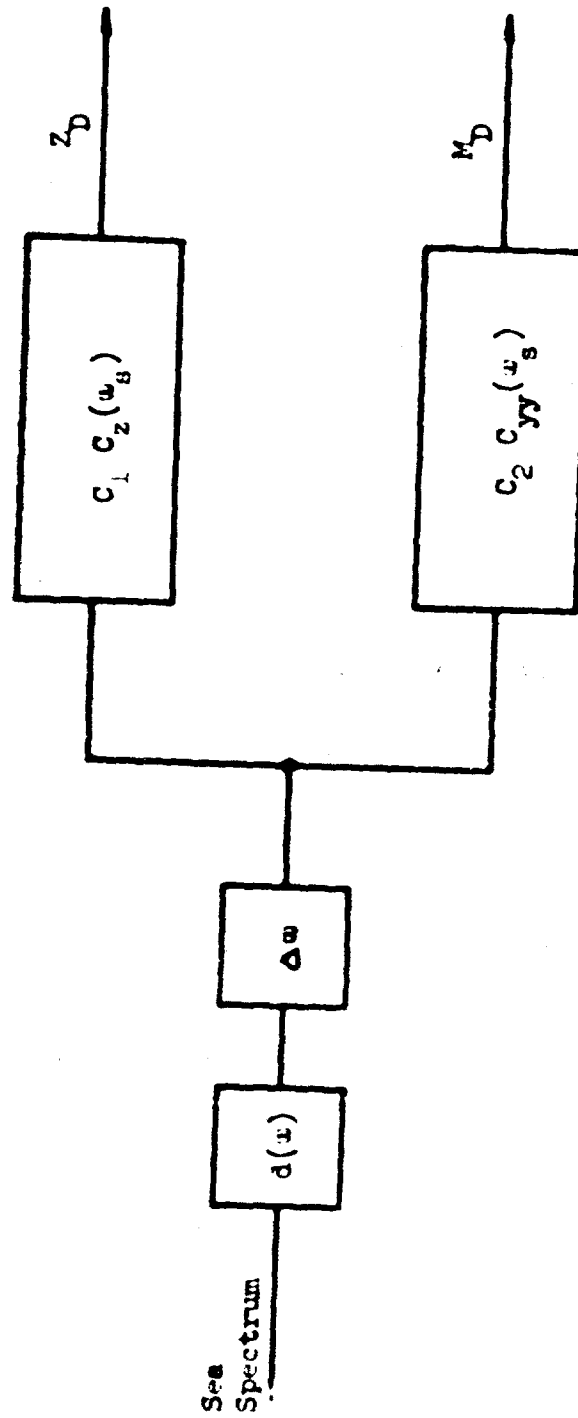
The operation ($\Delta \omega$) represents the apparent frequency shift in the sea spectrum due to the fact that the submarine is moving with respect to the sea. The function $d(\omega)$ represents the attenuation due to depth for waves of various frequencies. The term $C_1 C(\omega_s)$ and $C_2 C_{yy}(\omega_s)$ are the transfer functions developed by Havelock.

The frequency shift ($\Delta \omega$) can be determined quite readily. The speed of propagation of a wave (c) is only a function of its wavelength (λ)[46],

$$c^2 = \frac{g\lambda}{2\pi} .$$ (31)

The frequency of this wave as observed by a stationary object is:

$$\omega = \frac{2\pi c}{\lambda} .$$ (32)

The frequency of this wave as observed by an object which is moving at a velocity V_o in the direction directly opposite to the wave propagation direction ($\psi_w = 0$) is:

$$\omega_s = \frac{2\pi}{\lambda} (V_o + c) .$$ (33)

Combining equations (31), (32), and (33) gives a direct relationship between the sea spectrum frequency (ω) and the frequency observed by the submarine (ω_s),

$$\omega_s = \omega(1 + \frac{V_o \omega}{g}).$$ (34)

- 82 -

This relationship allows the power spectral density of the sea to be expressed as a function of the frequency ω_s. Remembering that,

$$S^2(\omega) \, d\omega = S^2(\omega_s) \, d\omega_s \tag{35}$$

the new power spectral density is,

$$S^2(\omega_s) = \frac{S^2(\omega)}{1 + \frac{2V_o \omega}{g}} \tag{36}$$

The depth attenuation of the effects of the various frequency waves is expressed as,

$$d(\omega) = \frac{\omega}{c} \, e^{-\frac{\omega d}{c}} \tag{37}$$

where   d   is the depth.

Havelock determined the functions $C_z(\omega_s)$ and $C_{yy}(\omega_s)$ to be,

$$C_z(\omega_s) = \frac{3}{2\pi e^{3/2}} \sqrt{2} \left\{ \left[1 + K_2 + f\left(\frac{2\pi L}{\lambda}\right)^{\frac{1}{2}} (K_2 - K_1) \right] \right.$$

$$\left. \left(\frac{\Delta}{L}\right)^{3/2} J_{3/2} \left(\frac{\pi e L}{\lambda}\right) \right\} \quad , \tag{38}$$

$$C_{yy}(w_s) = \frac{3\sqrt{2}}{2\pi e^2} \left(\frac{\lambda}{L}\right)^{3/2} \left\{ \left[1 + \left(\frac{\beta^2 + 1}{\beta^2 - 1}\right)^2 K' \right] J_{5/2}\left(\frac{\pi e L}{\lambda}\right) \right.$$

$$+ f \left(\frac{2\pi L}{\lambda}\right)^{\frac{1}{2}} \left[1 + \left(\frac{\beta^2 + 1}{\beta^2 - 1}\right)^2 K' \right.$$

$$\tag{39}$$

$$- \frac{2}{3} (1 + K_1)(1 + K_2) \quad J_{5/2}\left(\frac{\pi e L}{\lambda}\right)$$

$$\left. \left. + \frac{1}{3} (1 + K_1)(1 - 2K_2) \; J_{1/2}\left(\frac{\pi L}{\lambda}\right) \right] \right\},$$

where $\quad$ $J_{1/2}(\;)$, $J_{3/2}(\;)$ and $J_{5/2}$ are standard Bessel functions

L is the length of the spheroid $= 2a$

K_1, K_2 and K' are the axial, transverse and rotational
virtual inertia coefficients

a is one-half the length

b is one-half the maximum width

β is the length to beam ratio $= a/b$

- 84 -

e is the eccentricity $= (1 - \dfrac{b^2}{a^2})^{\frac{1}{2}}$.

The additional two constants C_1 and C_2 are defined as,

$$C_1 = \frac{4}{3} \pi g \rho \, a \, b^2,$$

(40)

$$C_2 = \frac{4}{3} \pi g \rho \, a \, b^2 \, L,$$

(41)

where ρ is the water density.

These relationships were combined numerically to produce the power spectral densities $H_z^2(\omega_s)$ and $H_{yy}^2(\omega_s)$ shown in Figures 24 and 25.

The above relationships developed by Havelock cannot be used for roll calculations. Therefore, for the sake of expediency in determining the roll spectrum, it is assumed that the body is square in cross section. For this case, it is also assumed that the sea is coming from directly abeam (the worst case) and that V_o is zero. This produces the approximate power density spectrum shown in Figure 26.

In order to be of any value in a linear analysis, these approximate spectra must be now approximated by some type of linear filter. A second order filter of the type,
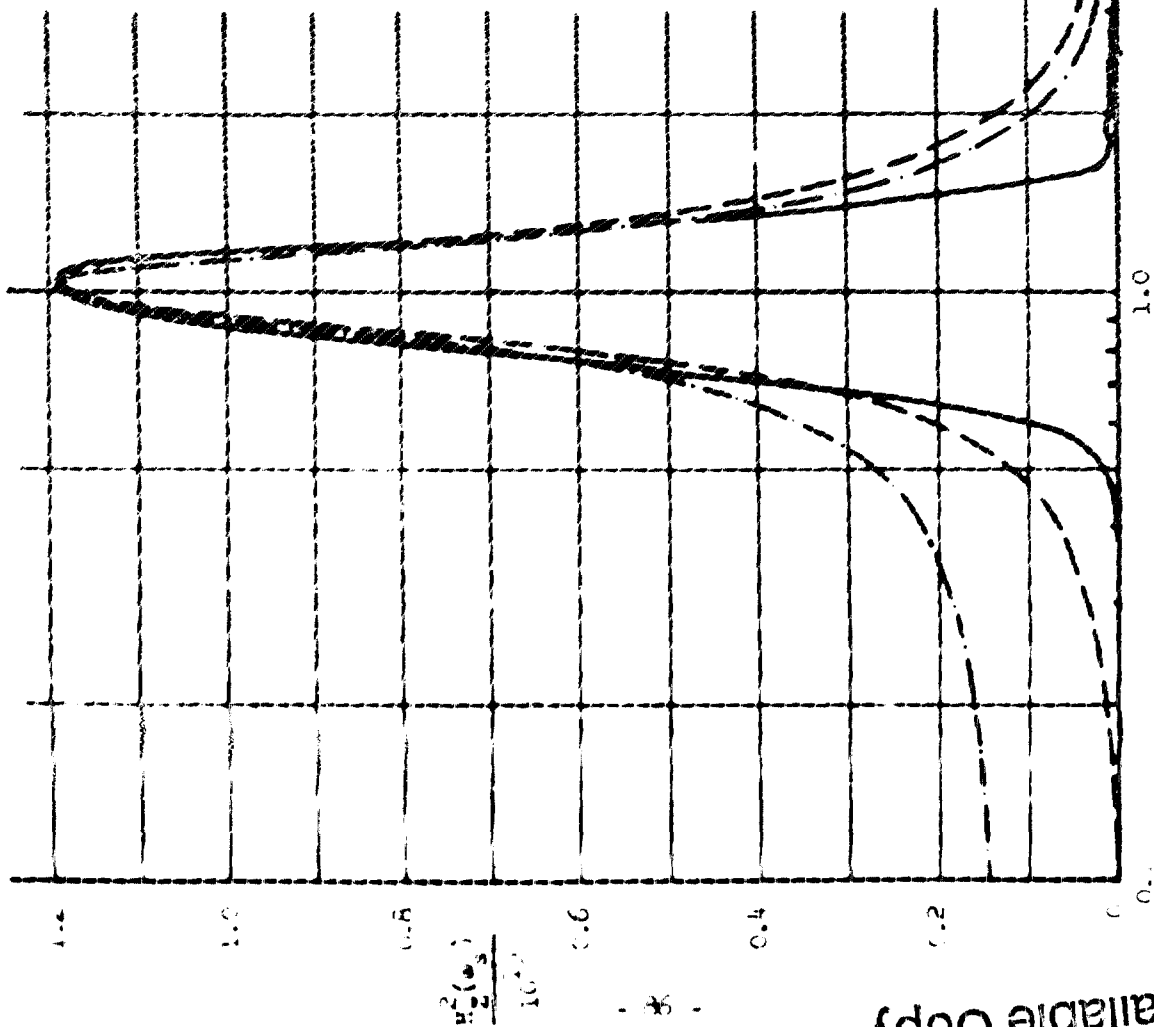
Figure 24

Shore Forcing Function Model

— For a Prolate Spheroid with:

Length = 400 ft
Width = 40 ft
Velocity = 40 ft/sec
Wind Velocity = 40 ft/sec
Depth = 40 ft
Virtual Inertial Coefficients:
$k_1 = 0$
$k_2 = 1$
$k' = 1$

--- For Linear Model:

$$\frac{r_{21}(s + r_{11})}{s^2 + r_{11}s + r_{01}^2} \quad \text{lb}_m\text{-ft/sec}^3$$

$r_{21} = 1.295 \times 10^7$ lb$_m$-ft/sec^3
$r_{31} = 1.07$ rad/sec
$r_{01} = 1.07$ rad/sec
$r_{11} = 0.535$ rad/sec

--- For Second Linear Model: $(r_{31} = 0)$

$r_{21} = 1.775 \times 10^7$ lb$_m$-ft/sec^3
$r_{01} = 1.03$ rad/sec
$r_{11} = 0.515$ rad/sec

1.0   10.0

Figure 25

Pitch Forcing Function Model

—— For a Prolate Spheroid with:

Length = 400 ft
Width = 40 ft
Velocity = 40 ft/sec
Wind Velocity = 40 ft/sec
Virtual Inertia Coefficients:

$k_1 = 0$
$k_2 = 1$
$k' = 1$

— — — For Linear Model:

$$\frac{\Gamma_{22} \dot\theta}{s^2 + \Gamma_{12}\dot\theta + \Gamma_{02}}$$

$\Gamma_{22} = 5.17 \times 10^9$ lb$_m$-ft^2/sec^3
$\Gamma_{02} = 1.33$ rad/sec
$\Gamma_{12} = 0.665$ rad/sec

- 87 -
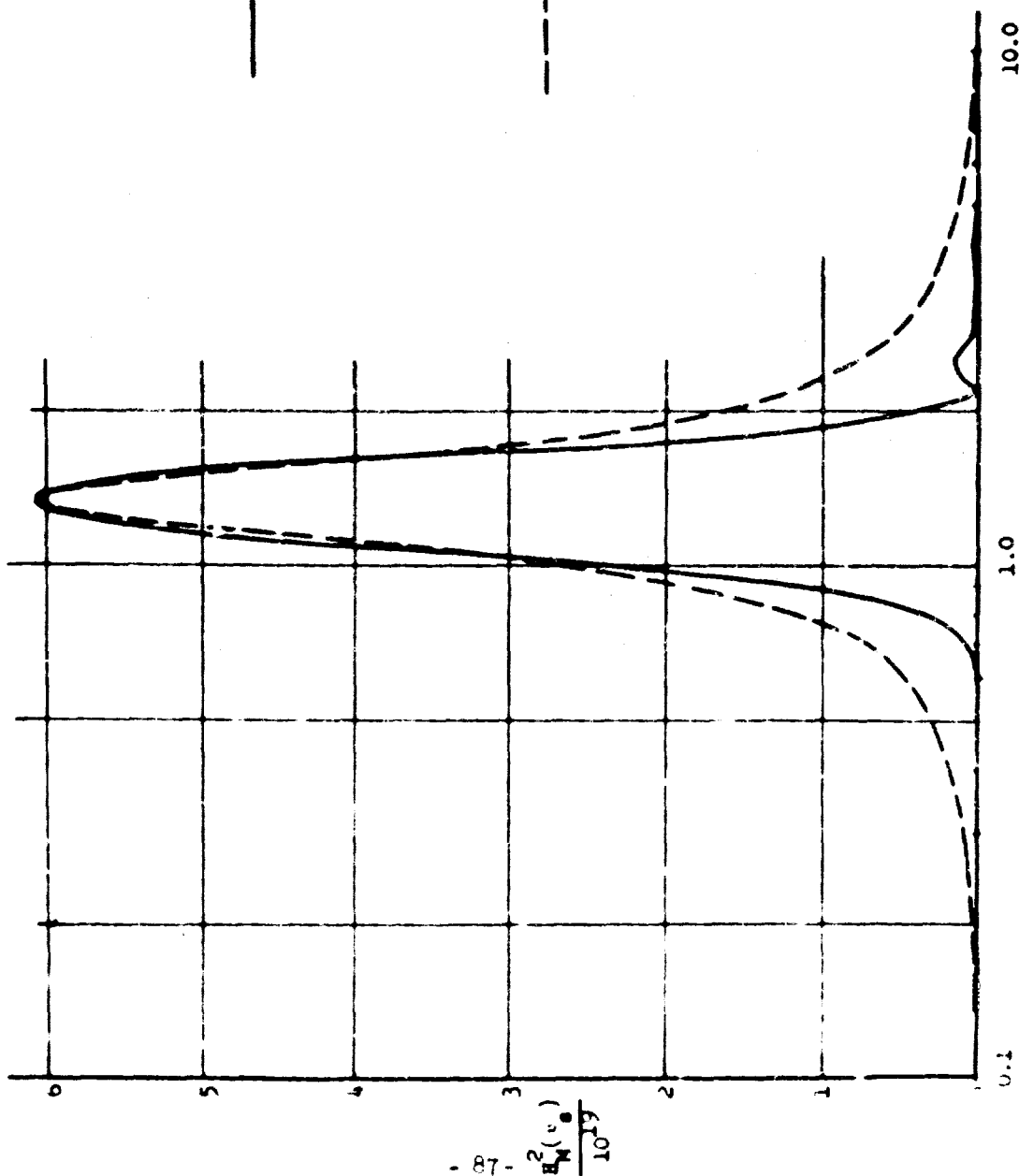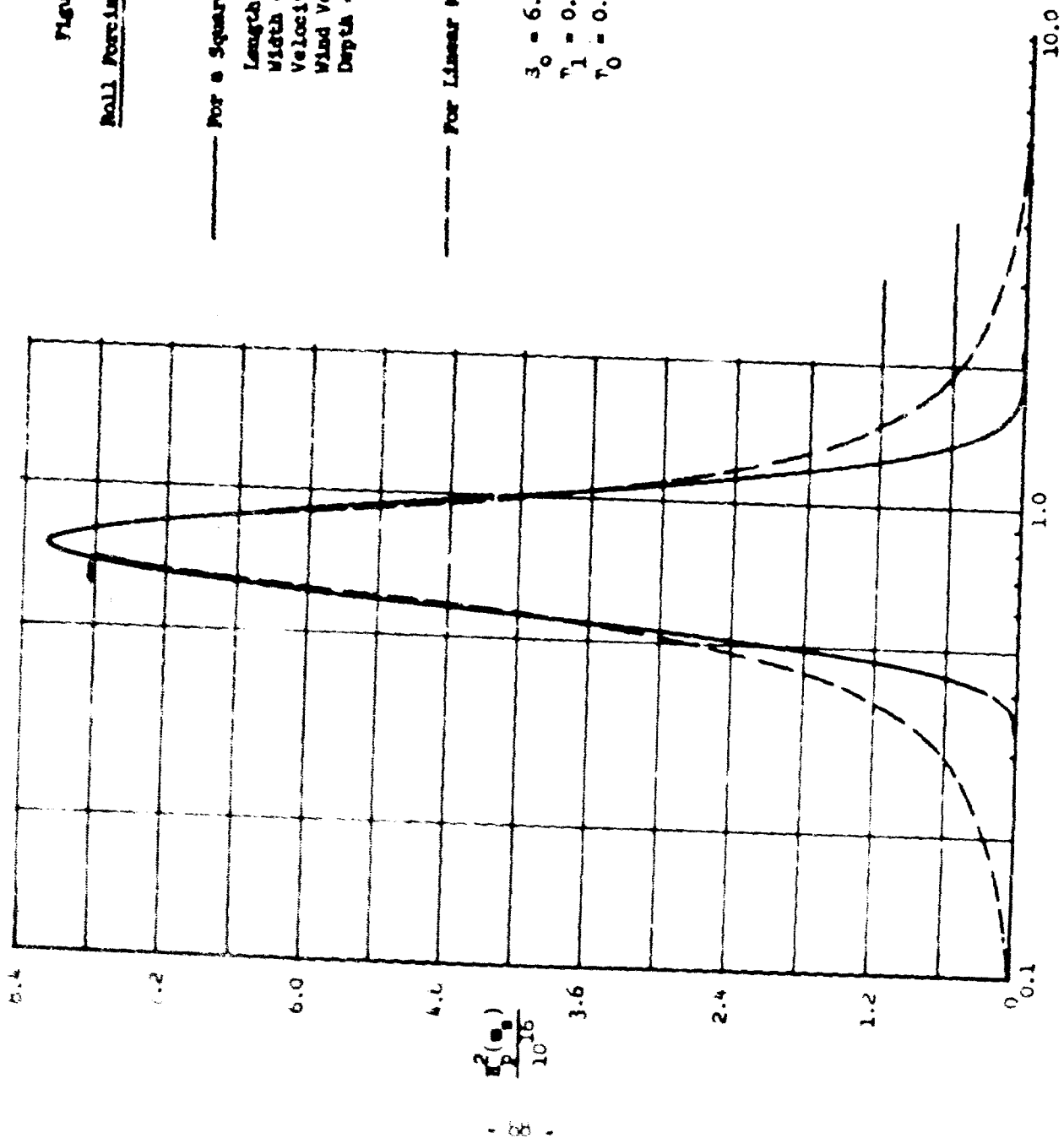
Figure 26

Roll Forcing Function Model

—————— For a Square Cross Section with:

Length = 400 ft
Width = 40 ft
Velocity = 0
Wind Velocity = 40 ft/sec
Depth = 40 ft

— — — — For Linear Model:

$$\frac{S_o}{s^2 + r_1 s + r_0^2}$$

$S_o = 6.04 \times 10^7 \ lb_m\text{-}ft^2/sec^3$
$r_1 = 0.438 \ rad/sec$
$r_0 = 0.730 \ rad/sec$

$$F(s) = \frac{\eta_{21} (s + \eta_{31})}{s^2 + \eta_{11}s + \eta_{C1}^2} \qquad (42)$$

was tried. Two filters are compared with the heave power density spectrum in Figure 24. The filter which sets η_{31} equal to zero appears to provide the best model. The filter with $\eta_{31} = \eta_{C1}$ was shown also because it has a special characteristic that some people feel is important. This is the fact that the autocorrelation function of this filter output is:

$$R_F(\tau) = C_F\, e^{-\frac{\eta_{11}}{2}|\tau|} \cos \left(\eta_{C1}^2 - \frac{\eta_{11}^2}{4}\right)^{\frac{1}{2}} \tau \qquad (43)$$

when the input to the filter is white noise. It has been generally suggested that the autocorrelation function of the ship disturbances could be best approximated by an exponentially-damped cosine function. It would be of interest to compare the output autocorrelation functions resulting from these two different filters.

For this study, the first filter ($\eta_{31} = 0$) was used as an approximation for all three forcing functions. (This filter produces a disturbance autocorrelation function which is the exponentially-damped sum of sine and cosine terms.)

The reduction of ship motions is certainly not a new subject[47]. However, it has only been within the past few years that people have become concerned with the optimal control of ship motions. As in many other fields of control, th emphasis has been placed on minimizing the mean square displacement. For instance, Davis[37] has shown that the minimum mean square error criterion leads to the "bang-bang" control of all control surfaces. He then went on to determine the switching criteria for this type of control. One only need envision a modern aircraft carrier sailing out to sea with its rudder flailing back and forth between its stops to wonder if it is really desirable to minimize the mean square error. In the next two sections we shall attempt to show that practical optimal control can be designed using the Max-Ranking Criterion.

5.2.1 Submarine Roll Control

The rolling action of a submarine can be modeled quite effectively as a simple second order system. This is because roll is so lightly coupled to the other five degrees-of-freedom. The system with roll control is shown in Figure 27.

The nomenclature which is used here agrees with that accepted by The Society of Naval Architects and Marine Engineers[48]:

I_x - moment of inertia about the roll axis

$K_{\dot{p}}$ - virtual moment of inertia about the roll axis

$X_{\dot{p}}$ - roll acceleration

X_p - roll velocity

- 90 -

Figure 27

Roll Control of a Submarine
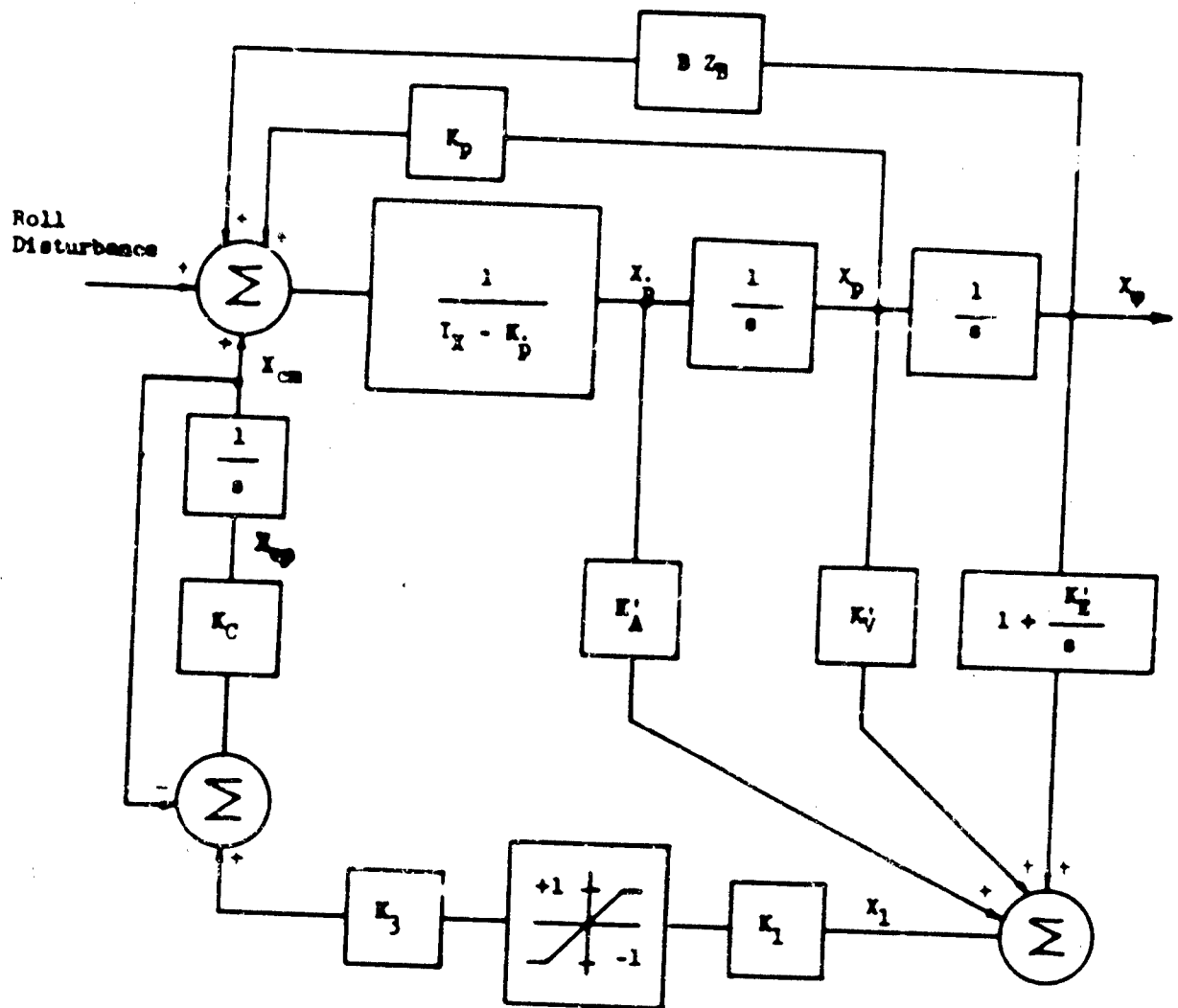
Figure 28

Roll Disturbance Model
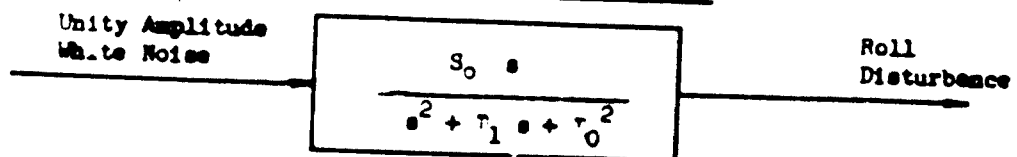
X_{φ} - roll angle

B - buoyancy

Z_B - metacentric height (must be negative for stability)

K_p - roll damping (must be negative for stability)

Values which are typical for modern submarines are used,

$$I_x = 3 \times 10^9 \text{ lbm-ft}^2$$

$$K_{\dot{p}} = -3 \times 10^8 \text{ lbm-ft}^2$$

$$BZ_B = -2.5 \times 10^9 \text{ lbm ft}^2/\text{sec}^2$$

$$K_p = -10^9 \text{ lbm ft}^2/\text{sec}$$

From Figure 26, the input disturbance is modeled by a unity amplitude white noise source and a second order filter as shown in Figure 28.

The values determined for the coefficients are:

$$S_o = 6.04 \times 10^7 \text{ lbm ft}^2/\text{sec}^3$$

$$\eta_o = .73 \text{ rad/sec}$$

$$\eta_1 = .438 \text{ rad/sec}$$

The control system chosen is about as general as is desirable. Provision is made for acceleration, velocity, displacement, and integral displacement feedback. The lag in the control path models the movement of the control surfaces in response to command signals. The variable, X_{cp}, is the time rate of change of the control moment, X_{cm}, and therefore

- 92 -

is a measure of instantaneous control power. The saturation nonlinearity serves to limit the control moment (X_{cm}) to a maximum value of K_3 and also restricts the control power (X_{cp}) to a maximum value of $2K_3K_c$. When the nonlinearity is replaced by the equivalent linear gain K_{EQ}, the five control parameters are then defined as :

$$K_D = \frac{K_1 K_{EQ} K_3}{B Z_B}$$

$$K_A = K_A' K_D$$

$$K_V = K_V' K_D$$

$$K_E = K_E' K_D$$

$$K_C = K_C$$

Four system variables are used in the optimization criterion. These are the roll acceleration $(X_{\ddot{p}})$, the roll angle error $(-X_{\varphi})$, the control moment (X_{cm}) and the control power (X_{cp}). The Max-Ranking Array which was used is shown in Table 5.

- 93 -

$J(1)$	$\sigma_{X_p}{}^2$ $(rad/sec^2)^2$	$\sigma_{X_\varphi}{}^2$ $(rad)^2$	$\sigma_{X_{cp}}{}^2$ $(lbm\text{-}ft^2/sec^3)^2$	$\sigma_{X_{cm}}{}^2$ $(lbm\text{-}ft^2/sec^2)^2$
0	0	0	0	0
1	6.48×10^{-5}	3.04×10^{-4}	10^{14}	1.6×10^{15}
2	2.59×10^{-4}	1.22×10^{-3}	9×10^{14}	6.4×10^{15}
3	1.04×10^{-3}	4.87×10^{-3}	4.9×10^{15}	2.56×10^{16}
4	6.48×10^{-3}	1.49×10^{-2}	2.56×10^{16}	1.03×10^{17}
5	2.59×10^{-2}	3.04×10^{-2}	4.10×10^{17}	4.10×10^{17}
6	10	10	10^{30}	10^{30}

Table 5

Max-Ranking Array

for Submarine Roll Control

These values are more meaningful in a different form. For example: $\sigma_X{}^2$ equal to 2.59×10^{-4} rad^2/sec^4 corresponds to an r.m.s. value of X_p which would produce a .01g acceleration at a 20 foot radius

(the maximum radius of the submarine hull); $\sigma_{X_\varphi}^2$ equal to 1.22 x 10^{-3} rad^2 corresponds to an r.m.s. roll angle of 2 degrees; $\sigma_{X_{cm}}^2$ equal to 6.4 x 10^{15} lbm-ft^2/sec^2 is the moment required to correct a 2 degree list of the submarine or to produce a .025 rad/sec^2 angular acceleration of the body; $\sigma_{X_{cp}}^2$ equal to 9 x 10^{14} corresponds roughly to the ability to move the control surface from the zero position to the position which produces the mean square moment $(\sigma_{X_{cm}}^2 = 6.4 \times 10^{15})$ in three seconds.

The last row in Table 5 was purposely made unrealistically high. This allows some measure of comparison during the early phase of the search when the mean square values are apt to be high. If this row were not present, the selection process would discard all systems which produced any values greater than those contained in row $J(1) = 5$. This, in effect, discards information which is of value to the search procedure and thus decreases the efficiency of the search. This added row cannot confuse the end results, since, for this case, any value of J greater than 5 is readily recognized as an unacceptable system.

The digital computer program for the problem was written using equivalent linearization and algebraic evaluation. This program is included in Appendix A.2. It should be noted that required computation for this sixth order system is considerably longer than for the previously discussed third order system, however, the search procedure and the use of the Max-Ranking Criterion are essentially unchanged.

This program was run several times with varying values of
the search exponent (m). After the first two short runs, it was
obvious that the integral control term (K_E) should be zero. The remain-
ng runs were made with only a four parameter search. The results of
hese runs are listed in Table 6.

The presence of K_E as a search variable lengthened and
omplicated the procedure. The reason for this is the fact that there
s only a narrow range of values of K_E which yield a stable system.
igure 32 shows this for the region near the minimum and also points
it the fact that K_E should be equal to zero for best results.

In runs number one and two, only 25 to 30 percent of the
ital choices were stable. Setting K_E equal to zero increased the
action of stable points to around 50%. This j still a low percentage,
d it suggests that either a high percentage of the search space is
stable, or a large portion of the search is being carried out in the
cinity of an unstable region. Figure 31 shows that the value of K_D
ich produces a minimum is, in fact, directly adjacent to an unstable
gion.

All six runs were within less than 2% of the lowest value
J_{min}, although some rather widely scattered values of the control
rameters were obtained. Runs three, four, and five produce close
eement and are all obviously near the same minimum. Runs one, two
l six may be approaching this same minimum, but the searches would
e to be continued to be sure.

- 96 -

Run Number	1	2	3	4	5	6
$J_{minimum}$	.685	.676	.669	.666	.666	.670
$J(1)$	.685	.676	.664	.666	.666	.670
$J(2)$	.233	.245	.226	.229	.227	.237
$J(3)$	.667	.675	.669	.666	.666	.670
$J(4)$	.049	.050	.060	.060	.060	.055
$\sigma_{X_A}^2 \times 10^5$	4.44	4.38	4.31	4.32	4.31	4.35
$\sigma_{X_V}^2 \times 10^5$	4.58	4.63	4.38	4.41	4.39	4.50
$\sigma_{X_D}^2 \times 10^5$	7.23	7.43	6.86	6.95	6.89	7.21
$\sigma_{X_{CF}}^2 \times 10^{13}$	7.89	8.01	9.58	9.62	9.60	8.71
$\sigma_{X_{CP}}^2 \times 10^{-13}$	6.77	6.75	6.69	6.66	6.66	6.70
$\sigma_{X_1}^2 \times 10^3$	4.93	4.24	.430	.421	.417	.483
K_A	5.20	8.05	1.99	1.90	1.97	3.39
K_V	1.03	-.009	.310	.248	.279	.125
K_D	-.557	-.919	-.976	-.951	-.995	-.911
K_B	.087	.009	←———— fixed at zero ————→			
K_C	.085	.055	.165	.170	.164	.114
7094 Computation time (min)	~1.5	~1.5	~1.4	~1.6	~1.3	~1.2
Exponent	3	5	3	5	7	7

Table 6

Submarine Roll Control Optimization Results

The region around the point determined by run five was examined and the results plotted in Figures 29 through 33. This form of computer printout turns out to be more useful than it was first thought. It proves to be an effective check on a number of items.

First, it provides absolute proof that a minimum has been reached. It, of course, does not prove that this is the true minimum and not just a relative minimum.

It can point out regions of instability near the design point. This is particularly helpful information, since any control in which small changes in a control parameter could cause instability, is not practical, let alone optimal.

Similarly, these figures indicate the sensitivity of the system to changes in the individual parameters. This, certainly, is also of interest to the designer.

Finally, it provides a check on the accuracy of the computational process. Inaccuracies caused by round-off errors will add a certain amount of scatter to the results.

Figures 29 through 33 show that the searching process was extremely accurate. The only control parameter which is not within 1% of the actual minimum is K_V (Figure 30). This is certainly excusable, since a $\pm$ 10% variation in K_V produces only a .02% change in the minimum. It should be pointed out that the search results obtained here are several times more accurate than are required in practice. The searches were lengthened for these studies, since searching procedure was also of interest.

Figure 29

J(1) Near Minimum

Figure 30

J(1) Near Minimum

Figure 31

J(1) Near Minimum

$K_{D_0} = -.995$

K_D/Z_{D_0}

Figure 32

J(1) Near Minimum

K_Z

Figure 33

J(1) Near Minimum

$K_{C_O} = .164$

$J(1)$

K_C/K_{C_O}
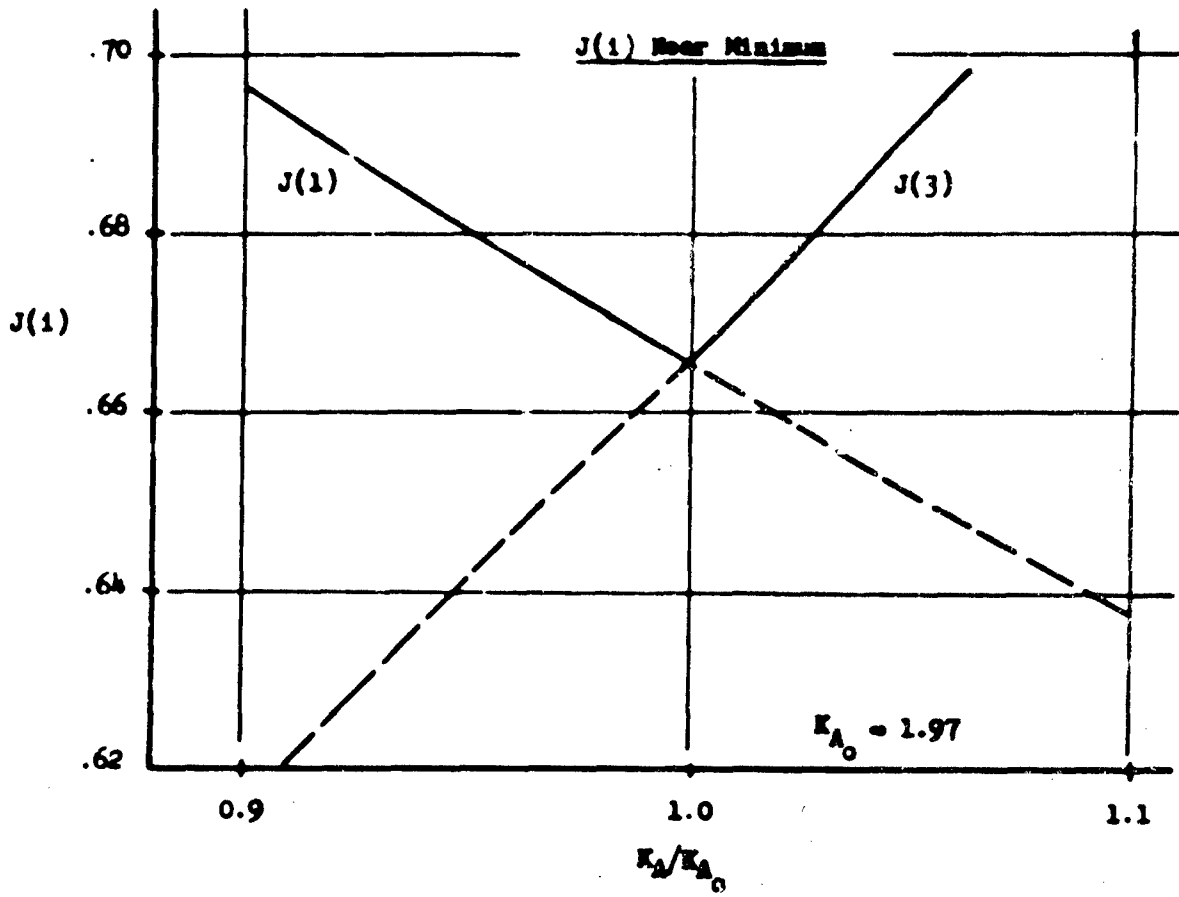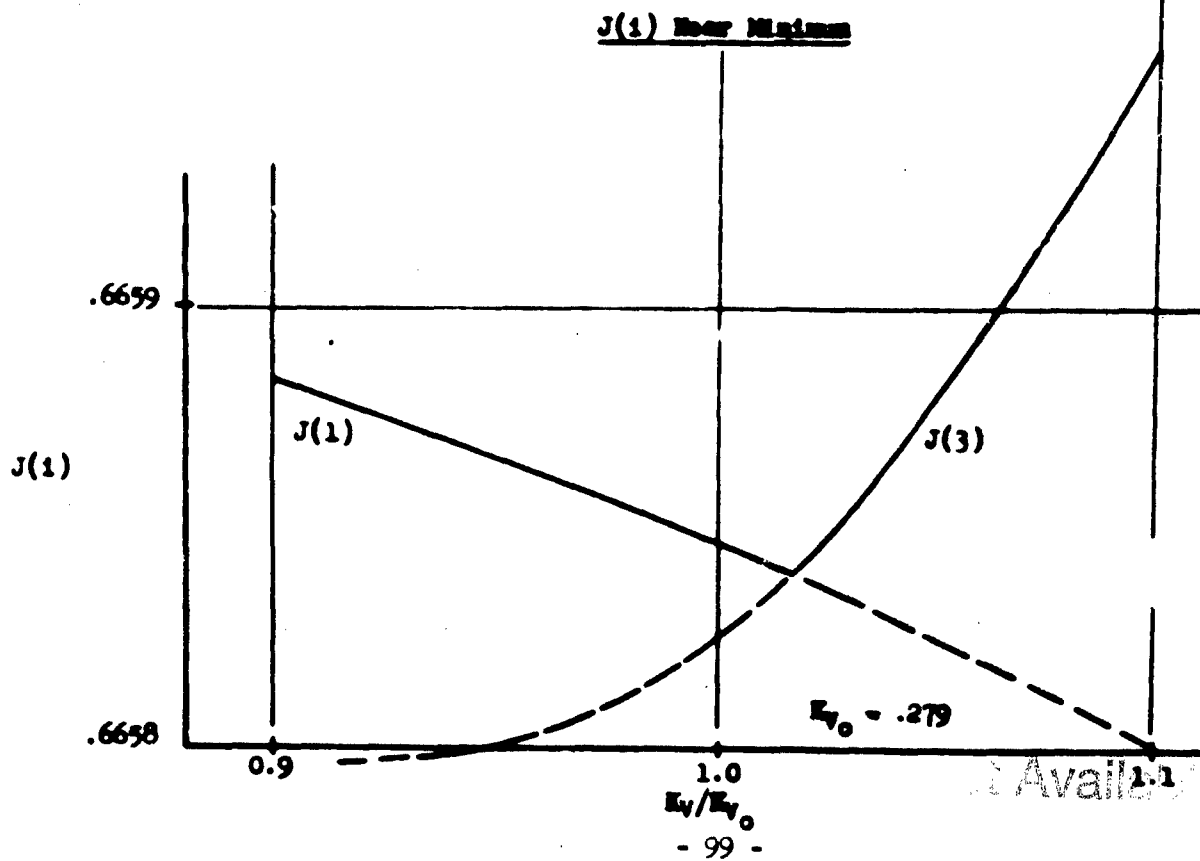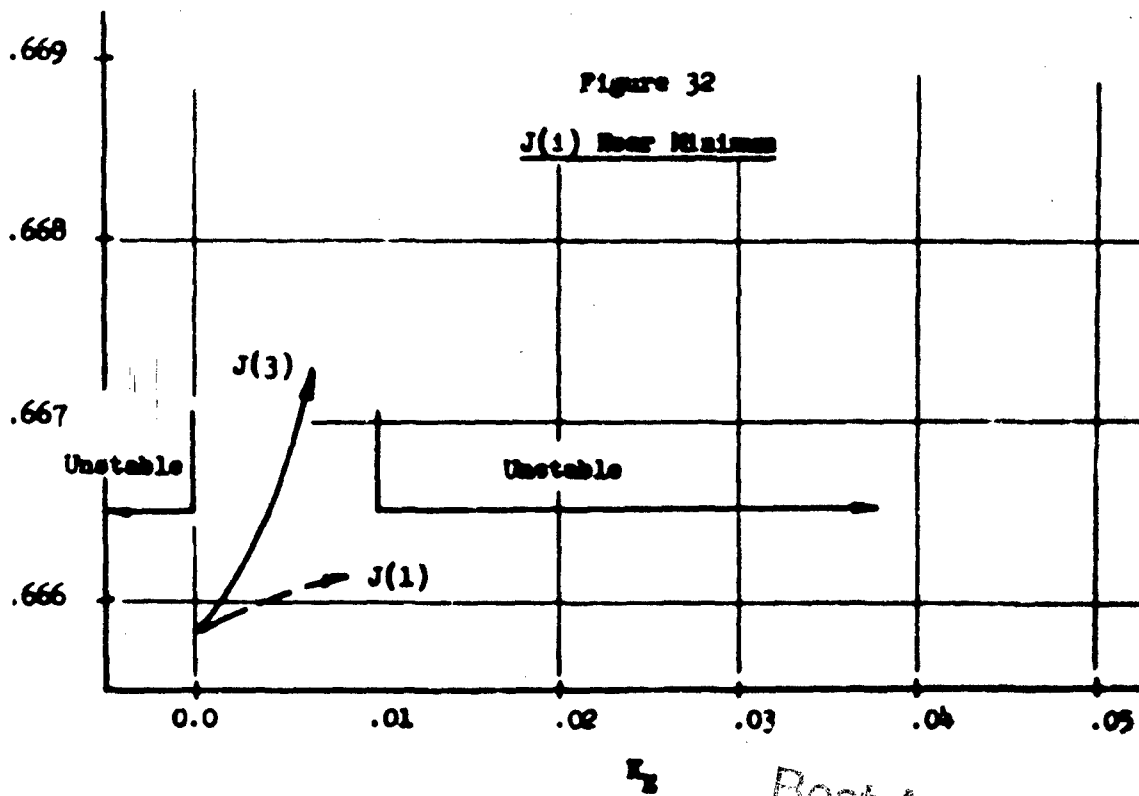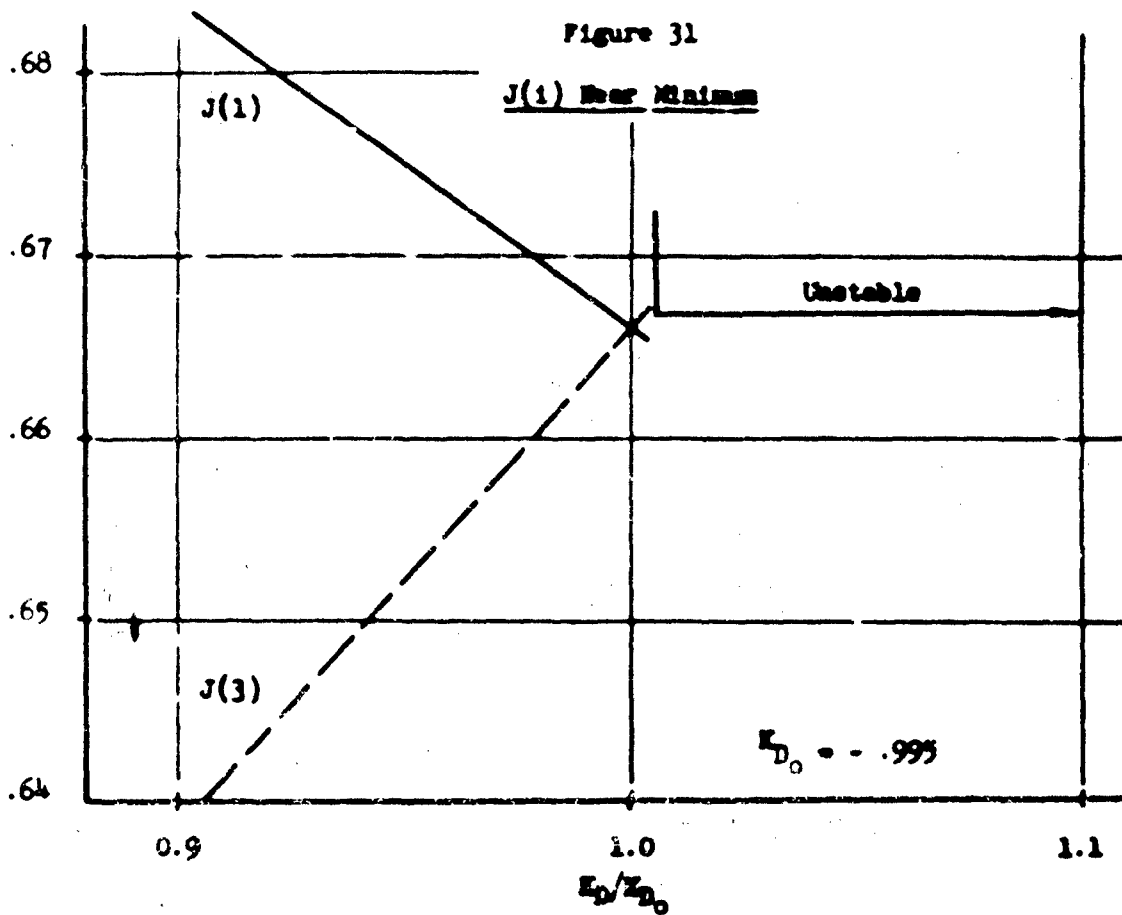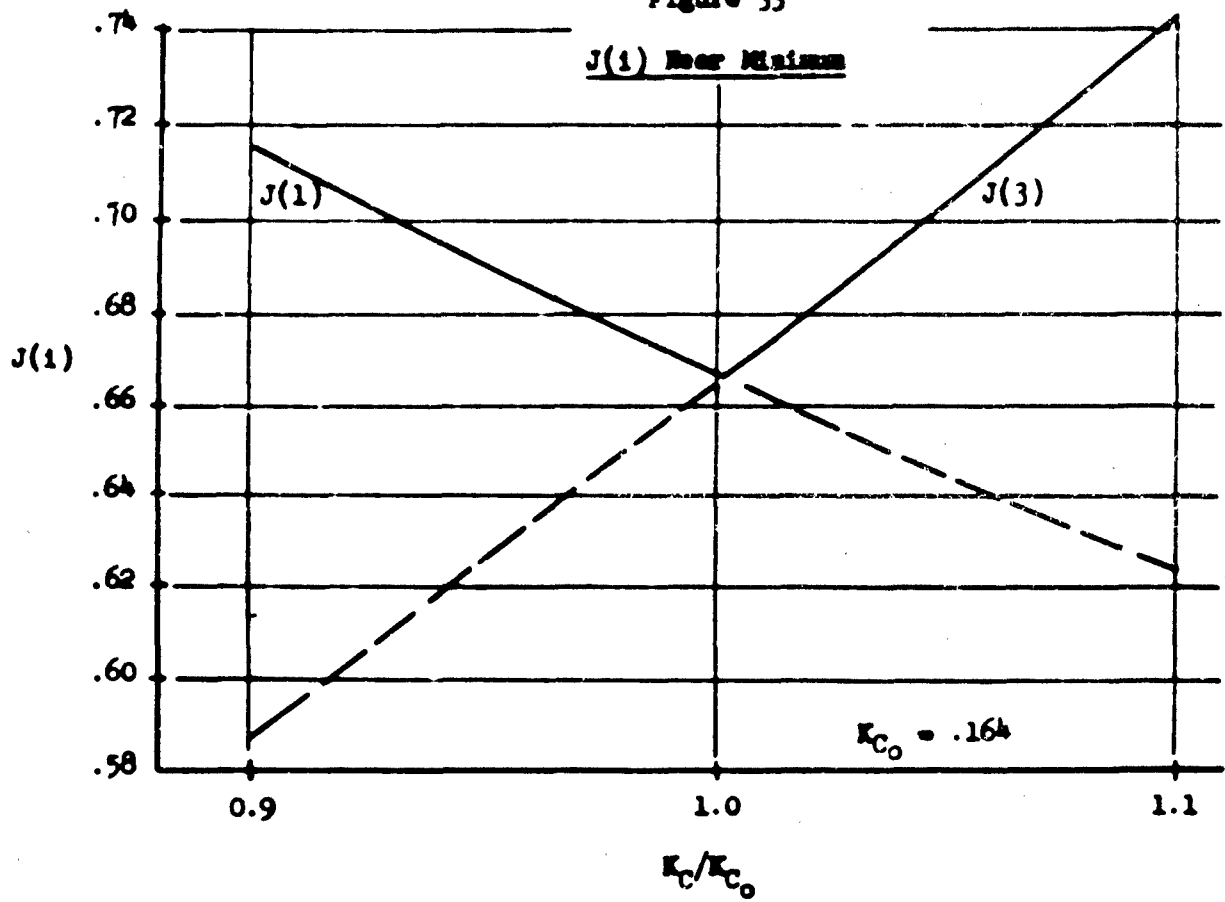
Figure 31 shows that the rinimum is immediately adjacent to a region of instability. However, it also shows that the control parameter K_D can be reduced by 10% while only reducing the system performance by 2.5%. If this is not desirable, an alternative is to fix K_D at some level (such as 0.90) and run a new three parameter search.

Using the values of $\sigma_{x_1}^2$ and τ_D in column four of Table 6, we can now express the gain K_1 (see Figure 27) as a function of the saturation limits, $\pm K_3$. This was obtained from Figure 6 using Booton's method of equivalent linearization. The results are shown in Figure 34.

Finally, it is of interest to see what affect the new control system has on the behavior of the system. This can be seen to a certain extent by examining the denominators of the controlled and uncontrolled systems. The uncontrolled system has a denominator equal to,

$$D_u = s^2 + 2 \zeta_u \omega_{nu} s + \omega_{nu}^2 \tag{44}$$

where

$\zeta_u = 0.174$, and

$\omega_{nu} = .87$ rad/sec

From Figure 26, we see that this is only slightly higher than the frequency corresponding to the maximum input moment ($\omega = .74$ rad/sec). The controlled system has a denominator of,

Figure 34

Gain (K_1) as a Function of the Limit (K_3)

Using Booton's Approximation
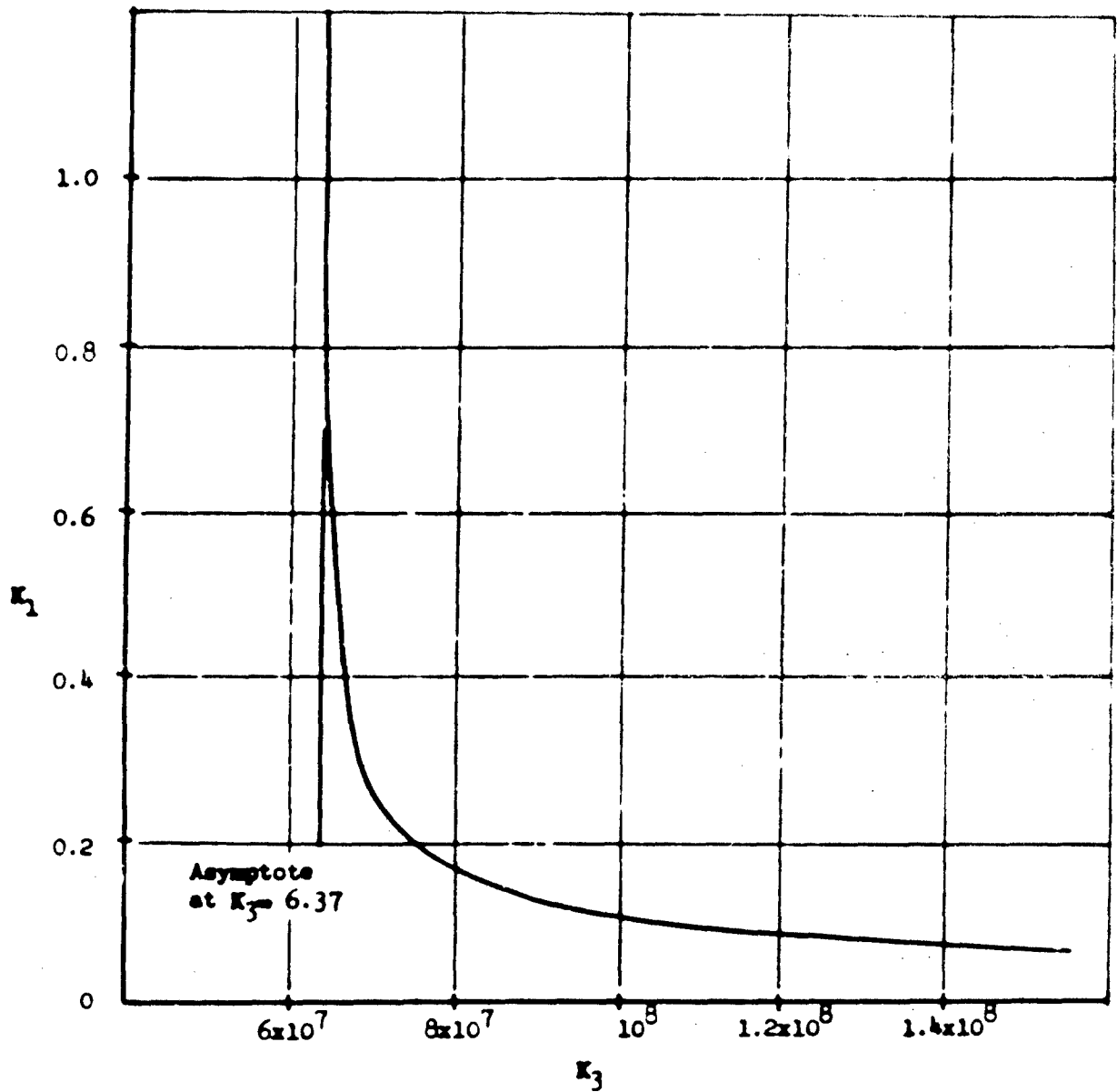
$$D_c = (s + a_1)(s^2 + 2\zeta_c \omega_{nc} s + \omega_{nc}^2)$$

(45)

.h

$a_1 = .0007 \text{ rad/sec.},$

$\zeta_c = .387, \text{ and}$

$\omega_{nc} = .92 \text{ rad/sec}$

.s, the control adds a new pole at $\omega = .0007$ rad/sec., shifts the
ural frequency to .92 rad/sec. and increases the damping ratio from
4 to .387. Each of these have the effect of reducing the response
the system to an input spectrum of the form of Figure 26.

.2 Submarine Pitch-Heave Control

The vertical motion (heave) and the vertical rotational
ion (pitch) of a submarine are very closely coupled. Thus, it is
practical to attempt to independently control these motions.
ure 35 shows the coupled pitch-heave system with the associated
trol system. The nomenclature used here agrees with the SNAME
ommendations[48]:

$\dot{w}$ — heave acceleration

w — heave velocity

$\dot{Z}_o$ — rate of change of depth

Z — depth

Z_c — desired depth

Figure 35

Pitch-Heave Control of a Submarine
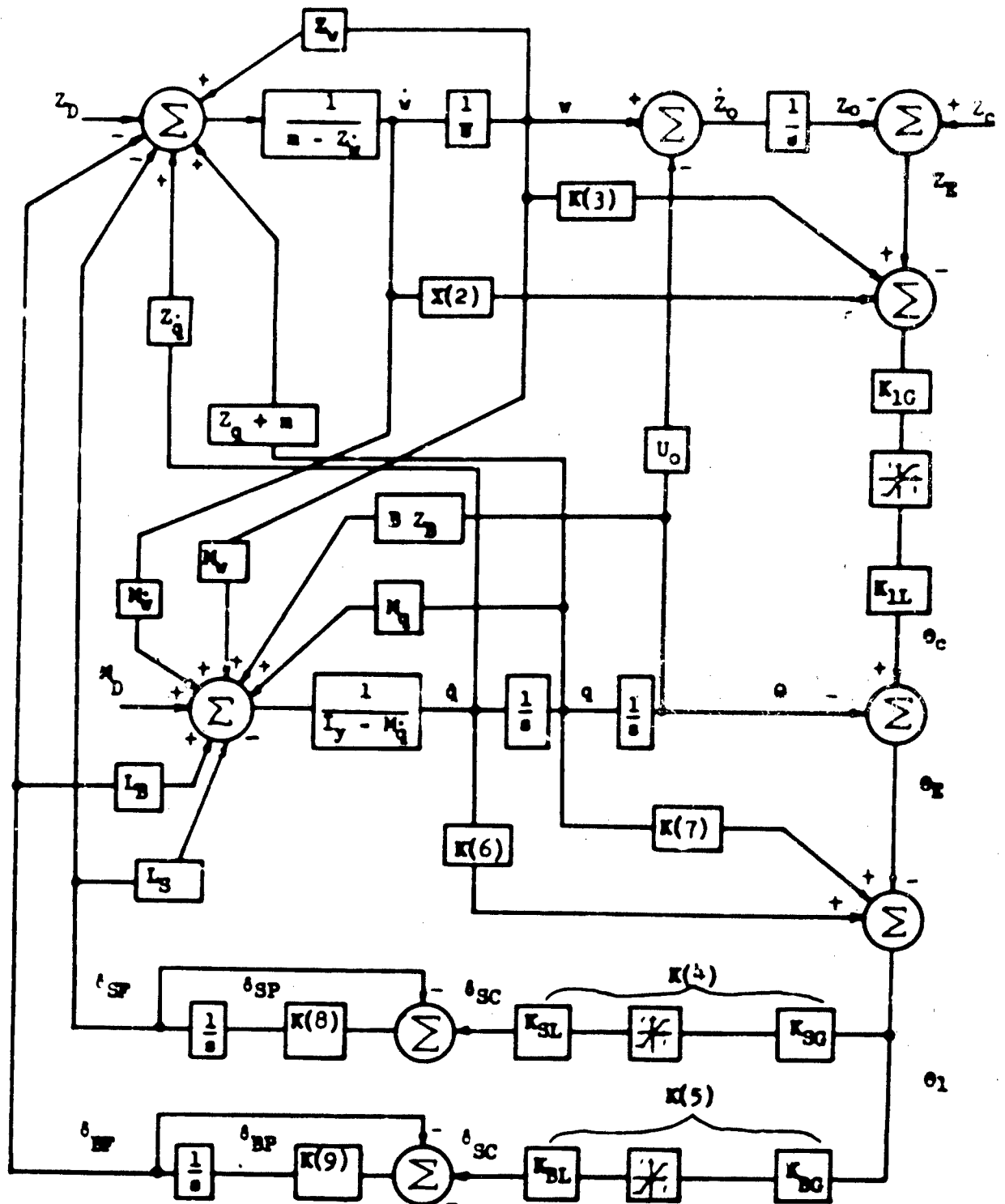
Z_E — depth error

$\dot{q}$ — pitch acceleration

q — pitch velocity

Θ — pitch angle

Θ_c — desired pitch angle

Θ_E — pitch angle error

δ_{SF} — stern plane force

δ_{SP} — stern plane power

δ_{BF} — bow plane force

δ_{BP} — bow plane power

Z_D — heave disturbance

M_D — pitch disturbance

m — submarine mass

$Z_{\dot{w}}$ — virtual mass in heaving

Z_w — heave dmaping

$Z_{\dot{q}}, Z_q$ — coupling coefficients relating pitching acceleration and velocity to heaving forces

I_y — moment of inertia in pitching

$M_{\dot{q}}$ — virtual moment of inertia in pitching

M_q — pitch damping

B — total submarine buoyancy (equal to mg for neutral buoyancy)

Z_B — metacentric height

$M_{\dot{w}}, M_w$ — coupling coefficient relating heaving acceleration and velocity to pitching moments

L_B — distance from the bow planes to the center of gravity

L_S — distance from the stern planes to the center of gravity

V_o — submarine forward velocity

K_1 to K_9 — control parameters

There are three saturating nonlinearities in this system. The one which limits the desired pitch angle will behave almost linearly for this type of disturbance. The purpose of this element is to prevent the desired pitch angle from exceeding the specified safety limits during sudden large changes in the desired depth.

The stern plane and bow plane control loops with their associated limiters are identical to the control loop employed for the roll control of the previous section.

The philisophy behind this type of submarine depth control is that changes in depth can be more easily accomplished by changing the submarines angle of attack than by attempting to maneuver it up and down with bow and stern plane forces.

The system values chosen for this study are:

$$m = 4 \times 10^7 \text{ lbm.}$$

$$Z_{\dot{w}} = -3.5 \times 10^7 \text{lbm.}$$

$$Z_w = -8 \times 10^6 \text{ lbm/sec}$$

$$Z_{\dot{q}} = -4 \times 10^8 \text{ lbm-ft}$$

$$Z_q = -8 \times 10^8 \text{ lbm-ft/sec}$$

$$I_y = 3 \times 10^{11} \text{ lbm-ft}^2$$

$$M_{\dot{q}} = -2.5 \times 10^{11} \text{ lbm-ft}^2$$

$$M_q = -2 \times 10^{11} \text{ lbm-ft}^2/\text{sec}$$

$$M_{\dot{w}} = -4 \times 10^8 \text{ lbm-ft}$$

$$M_w = 3 \times 10^8 \text{ lbm-ft/sec}$$

$$BZ_B = -2.5 \times 10^9 \text{ lbm-ft}^2/\text{sec}^2$$

$$L_S = 200 \text{ ft.}$$

$$L_B = 100 \text{ ft.}$$

$$V_o = 40 \text{ ft/sec}$$

The input disturbances were calculated previously in this chapter and are shown in Figures 24 and 25. The model used is shown in Figure 36.

This type of model complicates the future calculations somewhat, because the two input disturbances are not statistically dependent. Thus, the system output density spectra are dependent on the two input cross power density spectra as well as the two to power density spectra.

The values determined for the coefficients of these input filters are:

$$\eta_{21} = 1.775 \times 10^7 \text{ lbm-ft/sec}^3$$

$$\eta_{11} = .515 \text{ rad/sec}$$

$$\eta_{01} = 1.03 \text{ rad/sec}$$

$$\eta_{22} = 5.17 \times 10^9 \text{ lbm-ft}^2/\text{sec}^3$$

- 108 -

Figure 36

Input Disturbance Model for Coupled Pitch-Heave

$$\frac{\eta_{21} s}{s^2 + \eta_{11} s + r_{01}^2}$$

z_D

$$\frac{r_{22} s}{s^2 + \eta_{12} s + r_{02}^2}$$

M_D

Unity Amplitude
White Noise
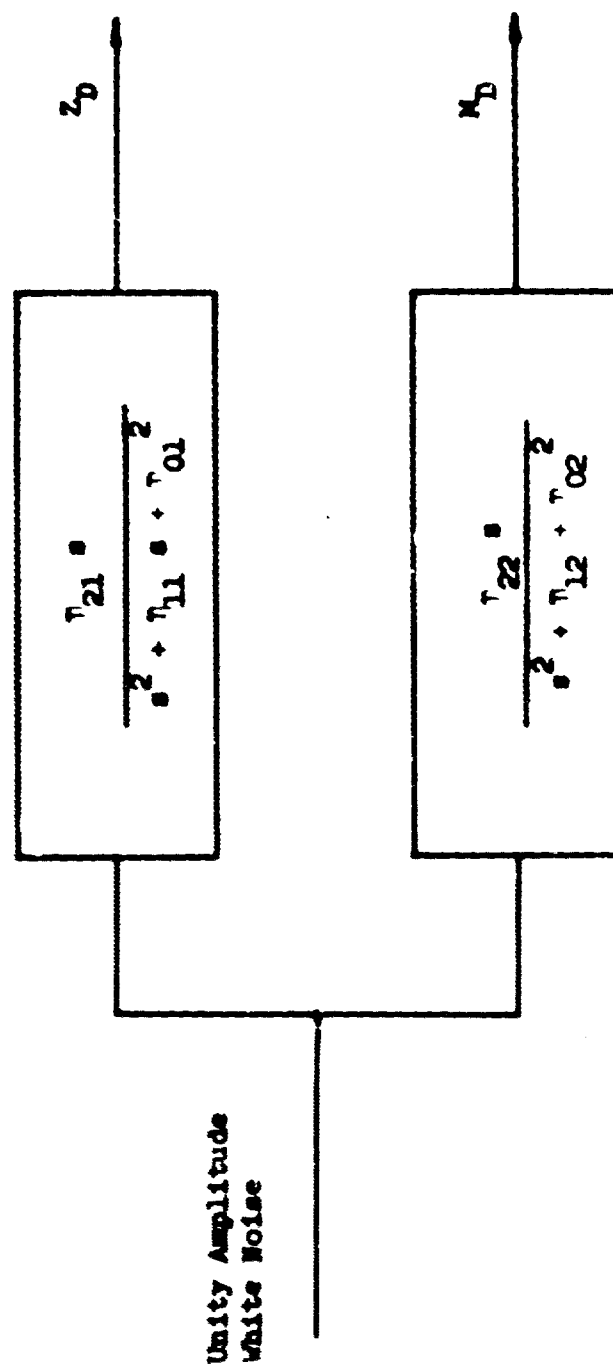
$$\eta_{12} = .665 \text{ rad/sec}$$

$$\eta_{02} = 1.33 \text{ rad/sec}$$

Eight system variables were used in the construction of the
ix-Ranking Array (Table 7). The values selected for the array are
ealistic specifications for present day submarines. As in the pre-
ious example, the final row was made very high to assist the search-
ig process.

The derivation of the numerous tenth order transfer functions
or this system was a monumental algebraic task. Provisions were made
or calculating the mean square values for twelve system variables.
a addition to the eight which occur in the Max-Ranking Array, values
or σ_w^2, σ_q^2, $\sigma_{z_1}^2$, and $\sigma_{\theta_1}^2$ were also determined. It is suggested that
or future work of this size, serious consideration should be given to
imulation techniques which require less algebraic manipulation but more
omputation time, or formula manipulation machine languages such as
ORMAC (an experimental Formula Manipulation Compiler currently being
eveloped by the IBM at the Boston Advanced Programming Department.)

The Fortran Computer program which was written for this pro-
lem is listed in Appendix A.2. The search and optimization procedures
re essentially the same as for the other two examples.

Finite difference integration was used to evaluate the mean
quare values, in contrast to the direct algebraic evaluation used in
he other two examples. This was required because of the inaccuracy
arried by round off errors in the algebraic approach (see Section 3.3.2).

Table 7

Max-Ranking Array for the

Coupled Pitch-Heave Control of a Submarine

J(1)	σ_w^2 $\left(\frac{ft}{sec}\right)^2$	σ_q^2 $(rad/sec^2)^2$	$\sigma_{Z_E}^2$ $(ft)^2$	σ_θ^2 $(rad)^2$	$\sigma_{\delta F}^2$ $(lbmft/sec^3)^2$	σ_{SF}^2 $(lbmft/sec^2)^2$	σ_{BP}^2 $(lbmft/sec^3)^2$	σ_{BF}^2 $(lbmft/sec^2)^2$
0	0	0	0	0	0	0	0	0
1	.0259	6.48×10^{-7}	.010	2.75×10^{-5}	2.50×10^{13}	4.00×10^{14}	6.25×10^{12}	10^{14}
2	.104	2.59×10^{-6}	.0225	7.61×10^{-5}	2.25×10^{14}	1.60×10^{15}	5.62×10^{13}	4×10^{14}
3	.415	1.04×10^{-5}	.040	3.04×10^{-4}	9.00×10^{14}	3.60×10^{15}	2.25×10^{14}	9×10^{14}
4	2.59	6.48×10^{-5}	.090	1.21×10^{-3}	4.90×10^{15}	1.00×10^{16}	9.00×10^{14}	2.5×10^{15}
5	10.4	2.59×10^{-4}	.250	2.75×10^{-3}	4.00×10^{16}	4.00×10^{16}	10^{16}	10^{16}
6	10^4	.100	100	1.0	10^{19}	10^{19}	10^{19}	10^{19}

or this case, finite a.fference integration worked very well. Since

oth input power spectra are narrow band, it was only necessary to

valuat the integral over the maximum range of the two spectra. The

esults were obtained using 25 increments in the range ω_f = .6 to

, = 2.2. A check on the accuracy was made by increasing the number

? increments to 50. This caused less than 0.1% change in the mean

[uare values obtained.

The search method was conducted a little differently for

is large system. It was estimated in Section 3.3.2 that approximately

)00 choices could be evaluated per minute for a 10th order system

ing finite difference integration. Equation (43) in Section 4.3

ves the estimate of the number of choices at about 64,000 assuming

at $(N_c)_1$ is 400. This indicates that it would require about one

ur of time on an IBM 7094 computer to produce one minimum which then

uld have to be checked.

To reduce computation time, it was decided to start several

)grams at different locations. They were run with search exponents

) of one and three. Each of these programs was allowed to run for

) minutes which is equivalent to 1500 to 2000 stable choices. The

ults of the initial runs were compared. Several new five minute

is were then initiated using starting points based on the earlier

ults. These decisions external to the computer prevent searching

a given area more than once and insure that no areas will be com-

tely neglected. Using this method, the total time required can be

to about thirty minutes.

The search space which was included and the final results of the search are listed in Table 8.

The nonlinearities can now be reinserted in the system. The equivalent gains K_4 and K_5 are replaced by K_{SL}, K_{SG}, and K_{BL}, K_{BG} respectively (see Figure 35). Figures 37 and 38 give K_{BG} and K_{SG} as functions of K_{BL} and K_{BL} respectively for the resulting values shown in Table 8. Since K_{1L} will be about 30 degrees (.524 radians), K_{10} will be equal to .010 for the results obtained here.

The system response was again calculated in the neighborhood of the minimum. The results of these calculations are shown in Figures 39 through 47.

These figures show that the search results were again more accurate than they needed to be. Unfortunately, this information is never known until after the fact. In reviewing the course of the search, it was determined that the last one-third of the search only produced an improvement of 1.5% in the system performance. This further points out the merit of first performing several crude searches when long total search times are expected.

Another very important fact pointed out by these figures is that a 4% change in any one of seven control parameters will cause instability. This behavior was expected after an early, two-minute, purely random search over the entire search space showed that 80% of the points considered produced instability.

- 113 -

Search Limits of Control Parameters	Final Values of Control Parameters	Final Values of Performance Indices	Final Values of Mean Square System Variables
$-.10 \leq K_1 \leq +.10$	.004895	$J(1) = .159$	$\sigma_w^2 = 4.12 \times 10^{-3}$
$-15 \leq K_2 \leq +15$	11.81	$J(2) = 2.54$	$\sigma_q^2 = 6.81 \times 10^{-6}$
$-20 \leq K_3 \leq +20$	-1.133	$J(3) = 2.38$	$\sigma_{z_\pi}^2 = 2.92 \times 10^{-2}$
$-10^{13} \leq K_4 \leq +10^{13}$	4.874×10^{12}	$J(4) = .158$	$\sigma_\theta^2 = 4.34 \times 10^{-6}$
$-10^{13} \leq K_5 \leq +10^{13}$	3.081×10^{12}	$J(5) = 2.54$	$\sigma_{SP}^2 = 5.90 \times 10^{14}$
$-5 \leq K_6 \leq +5$	1.661	$J(6) = .691$	$\sigma_{SF}^2 = 2.76 \times 10^{14}$
$-10 \leq K_7 \leq +10$	7.549	$J(7) = 2.54$	$\sigma_{BP}^2 = 1.48 \times 10^{14}$
$0 \leq K_8 \leq 30$	20.13	$J(8) = .750$	$\sigma_{BF}^2 = 7.50 \times 10^{13}$
$0 \leq K_9 \leq 20$	2.024		$\sigma_{z_1}^2 = 3.42 \times 10^3$
			$\sigma_{\theta_1}^2 = 1.17 \times 10^{-11}$
			$\sigma_w^2 = 5.10 \times 10^{-3}$
			$\sigma_q^2 = 4.75 \times 10^{-6}$

Table 8

Submarine Pitch-Heave Control Results

- 114 -

Figure 37

Gain (K_{DG}) as a Function of the Limit (K_{BL})

Using Booton's Approximation

- 115 -
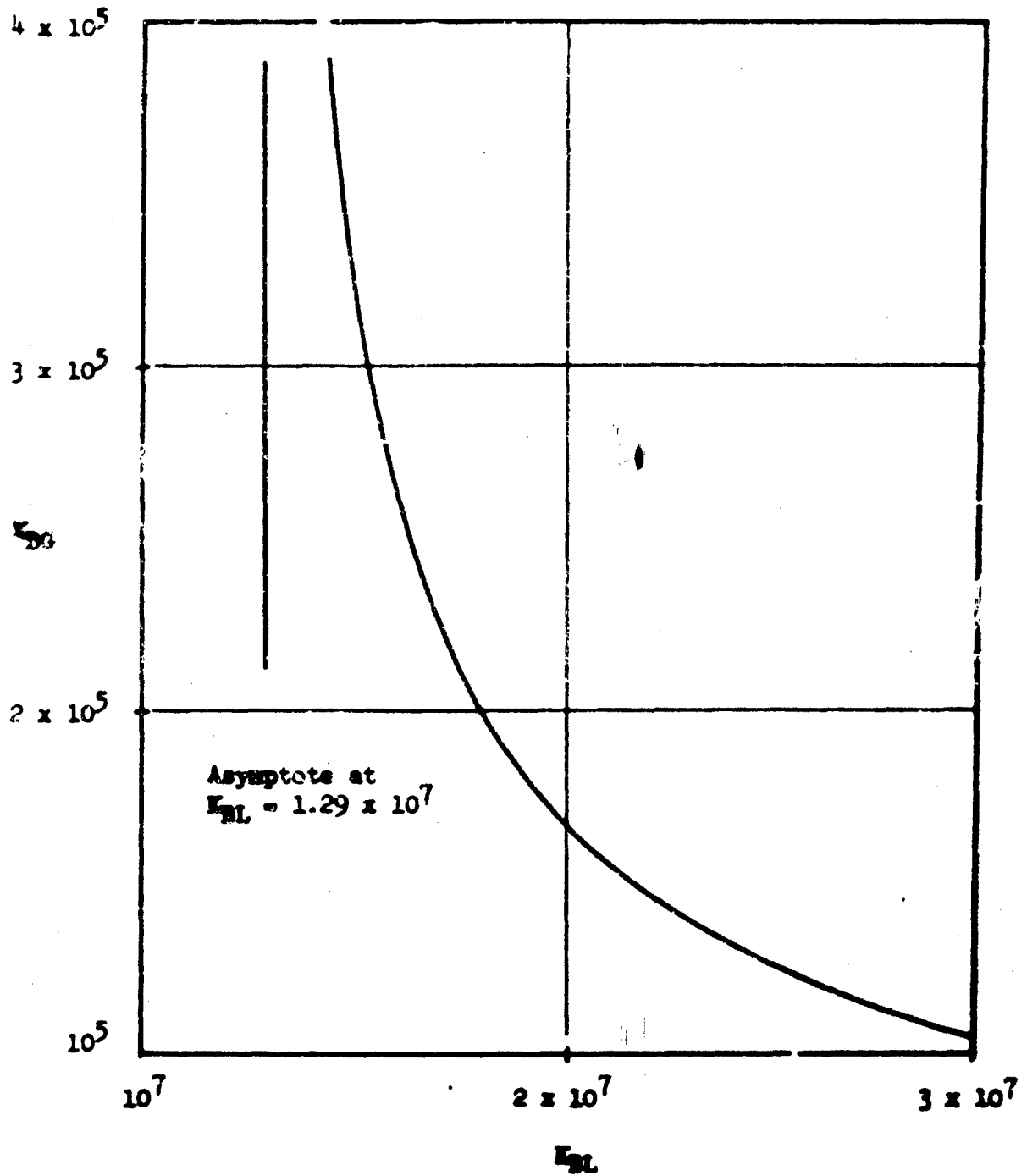
Figure 38

Gain (K_{80}) as a Function of the Limit (K_{SL})

Using Booton's Approximation

Asymptote at
$K_{SL} = 2.04 \times 10^7$
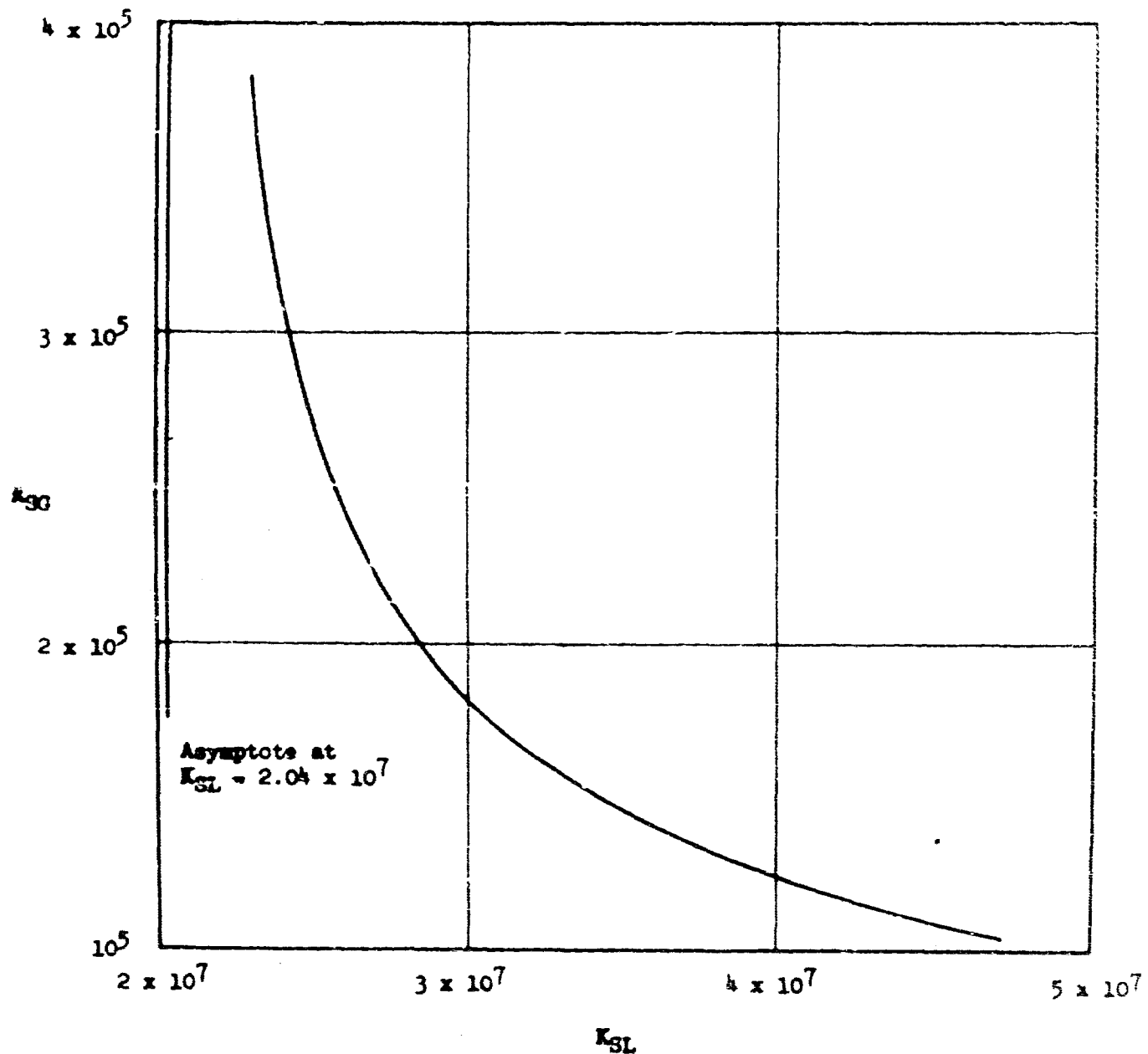
The problem of moving away from this region (or regions) of
instability can be very difficult. If each of the seven troublesome
control parameters is moved away from the unstable region simultaneouly,
there is no assurance that the resulting position is not near the same
or another unstable region, since no off-axes positions were checked
for stability.

It is doubtful that many other nine-parameter optimal con-
trols have ever been designed. Therefore, it is not readily apparent
whether this instability condition is universal for this number of
parameters or if it is just a peculiarity of this particular problem.
If it is a general problem, it would certainly be possible to test each
new minimum to assure that it is not within a given proximity to a
region of instability. This could be accomplished by a deterministic
or random scan of the surface of an n-dimensional hypercube which is
centered on the point in question.

Again, it is of interest to examine the denominators of the
transfer functions of the controlled and uncontrolled systems. The
uncontrolled system is unstable. The denominator of the transfer
function is:

$$D_u = s(s + .01622)(s + .0773)(s + .3893) \qquad (46)$$

The control system, first of all, must make the system stable. The
final system has a denominator of:

$$D_c = (s + .18)(s + 1.70)(s^2 + .097 s + .018)$$
$$(s^2 + 216 s + 366,800) \qquad (47)$$

The first, second-order term has an undamped natural frequency of .134 rad/sec and a damping ratio of .362. This is well below the principal disturbance frequencies (see Figures 24 and 25). The second, second-order term has an undamped natural frequency of 605 rad/sec and a damping ratio of .178. This is well above the principal disturbance frequencies.

Figure 39

J(1) Near Minimum

J(5) & J(7)

Unstable

J(2)

J(3)

$(K_1)_o = .004895$

$K_1/(K_1)_o$
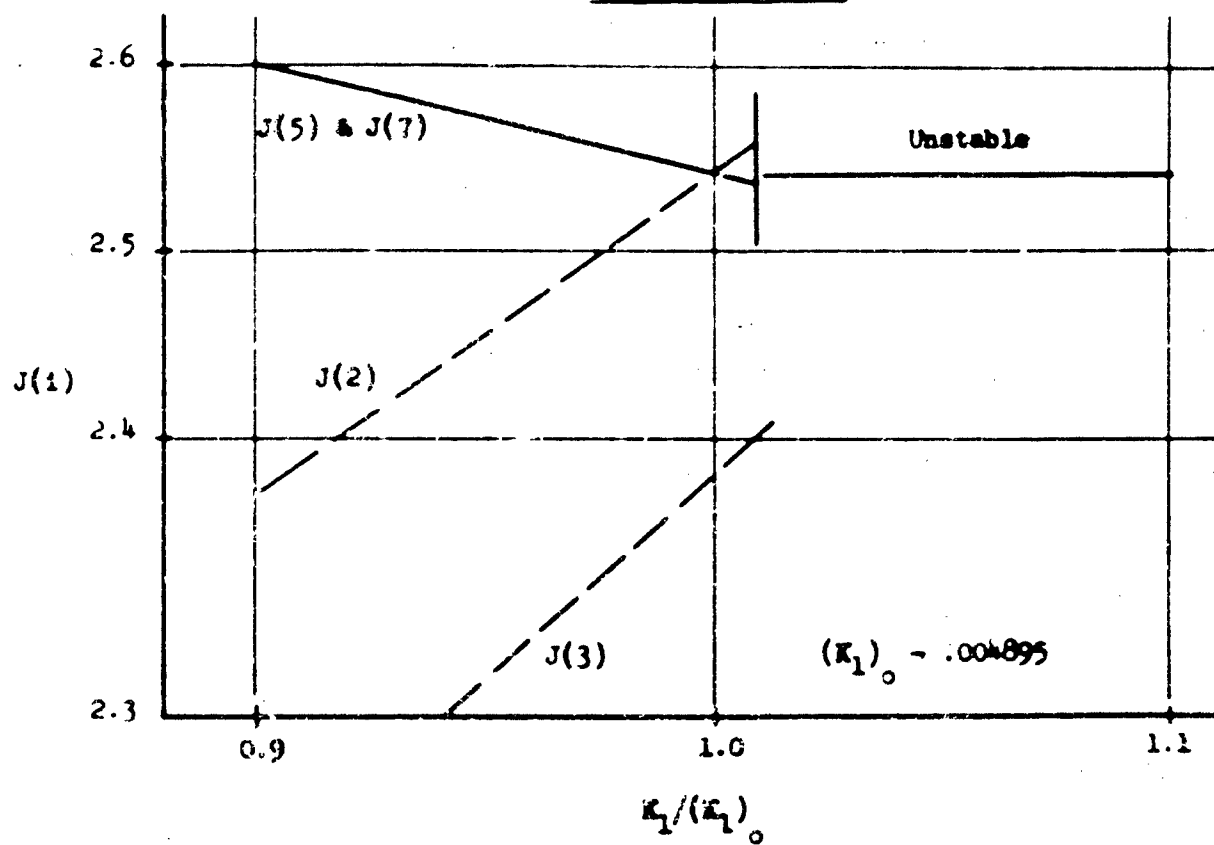
Figure 40

J(1) Near Minimum
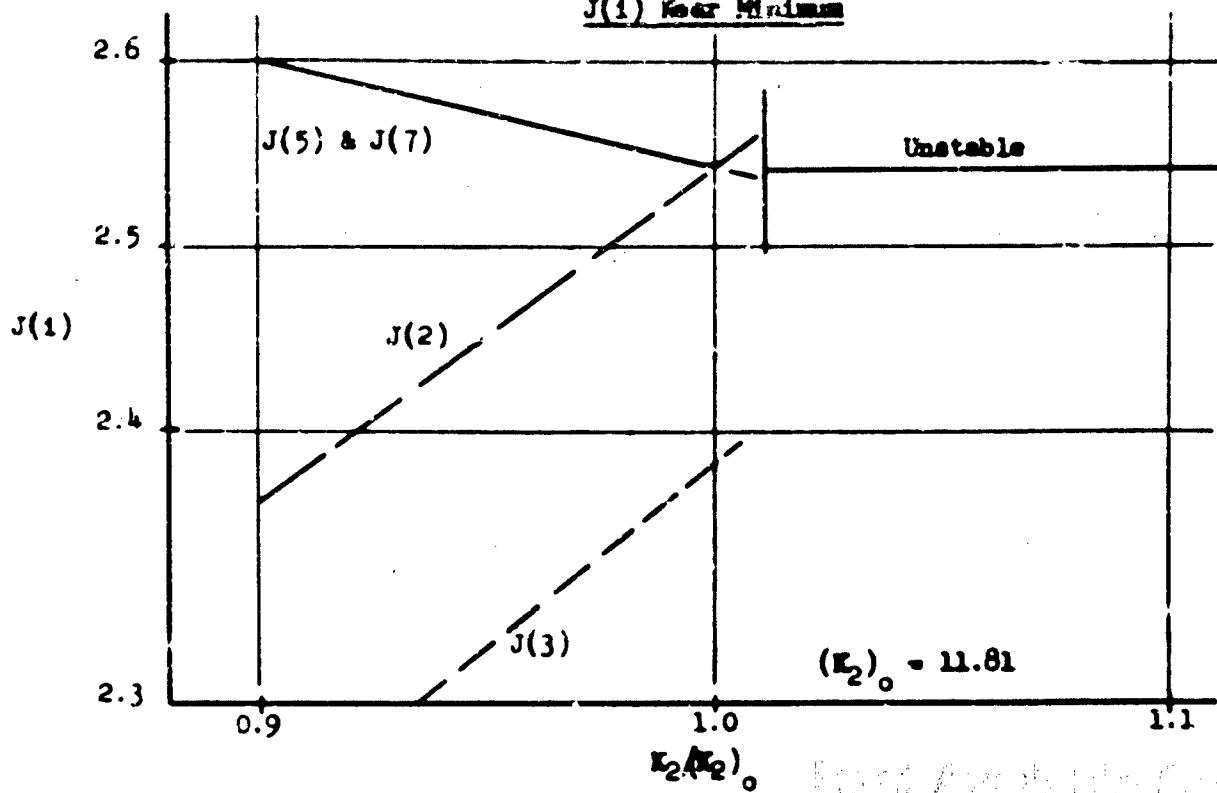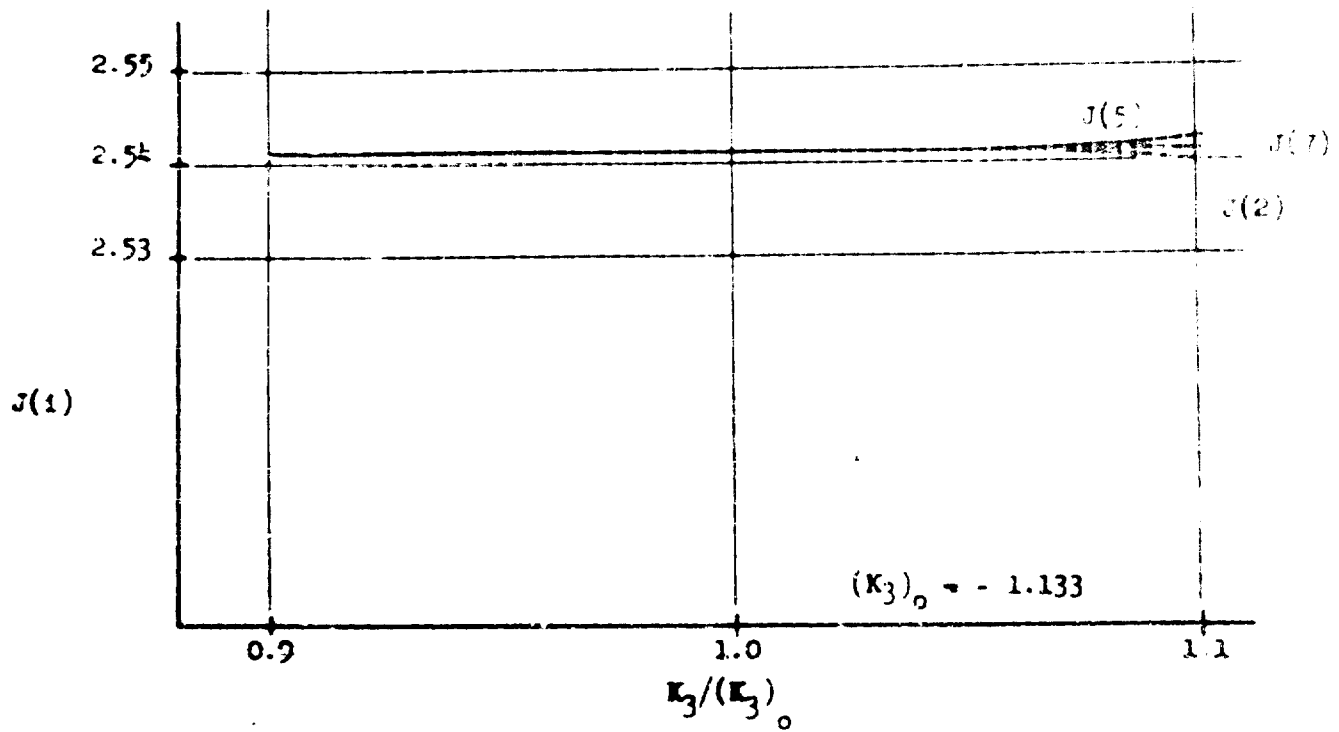
J(5) & J(7)

Unstable

J(2)

J(3)

$(K_2)_o = 11.81$

$K_2/(K_2)_o$

Figure 41

J(i) Near Minimum

$(K_3)_o = -1.133$

$K_3/(K_3)_o$

Figure 42

J(i) Near Minimum

Unstable

$K_4/(K_4)_o$

Figure 43

J(i) Near Minimum
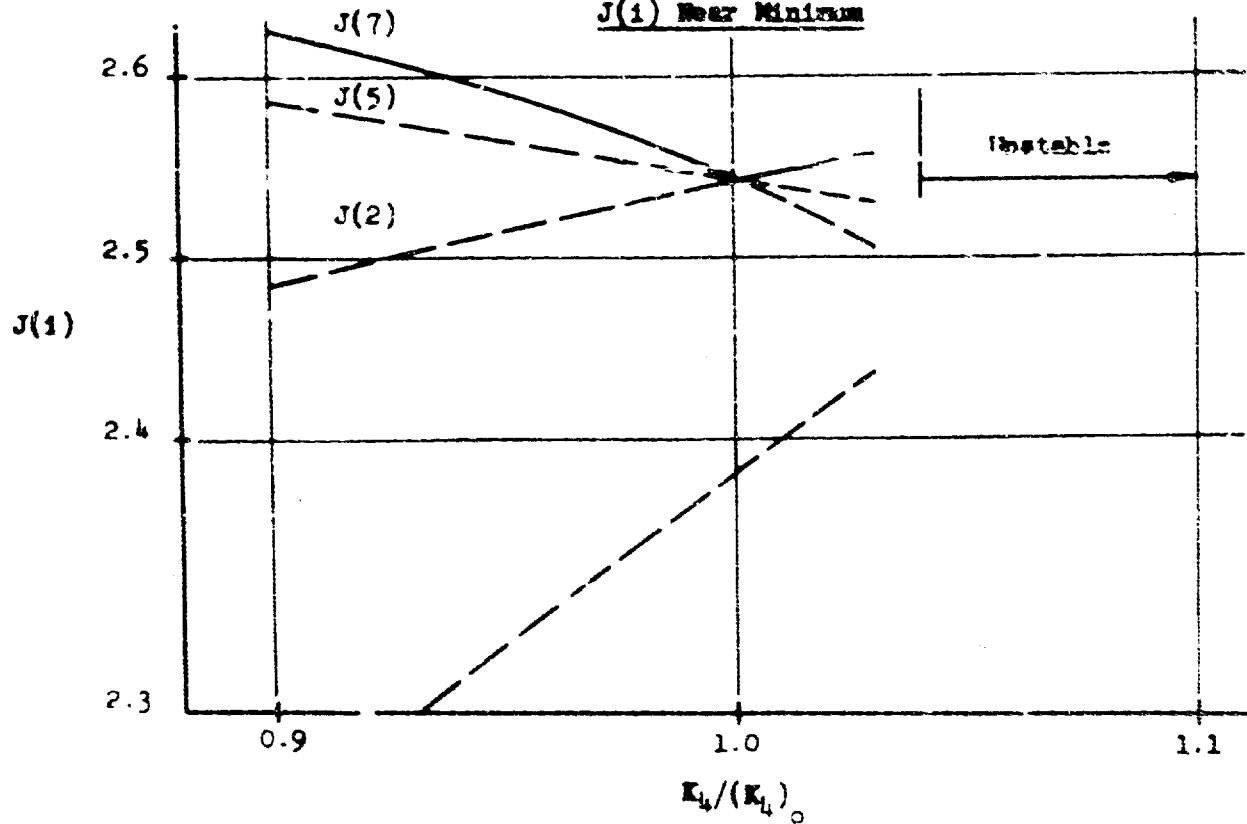
Figure 44
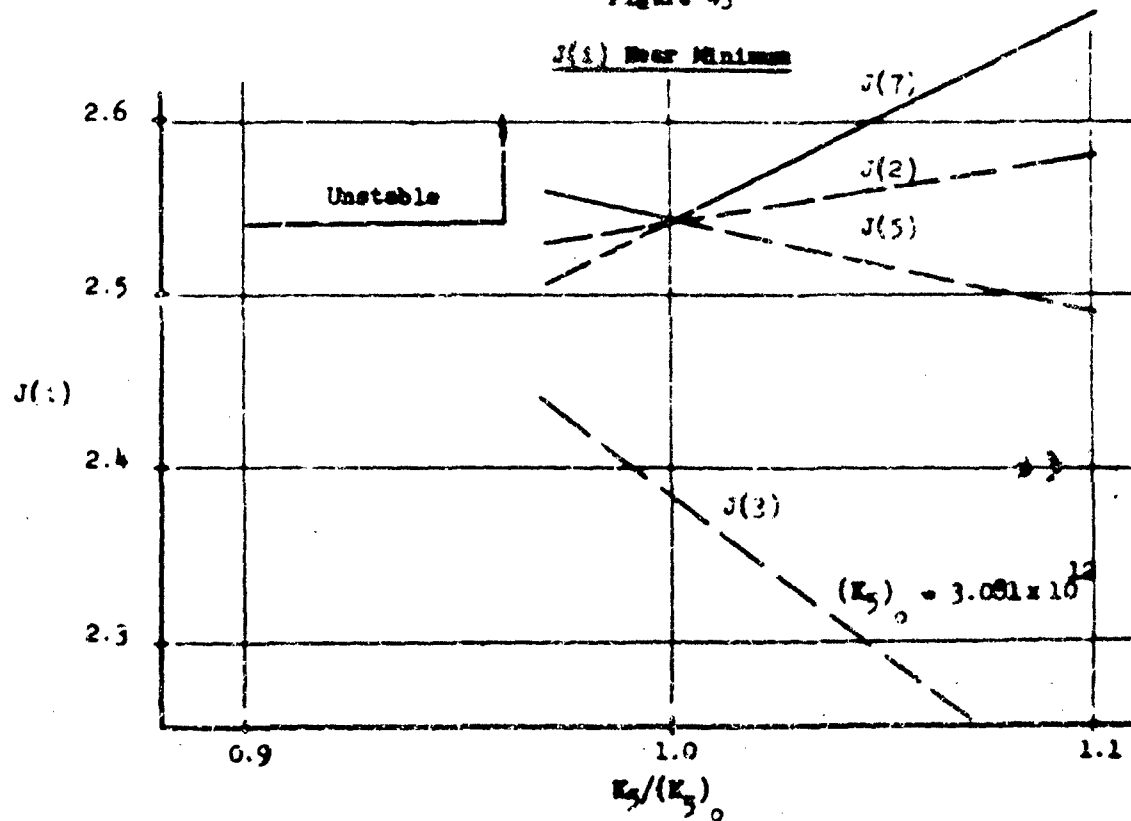
J(1) Near Minimum
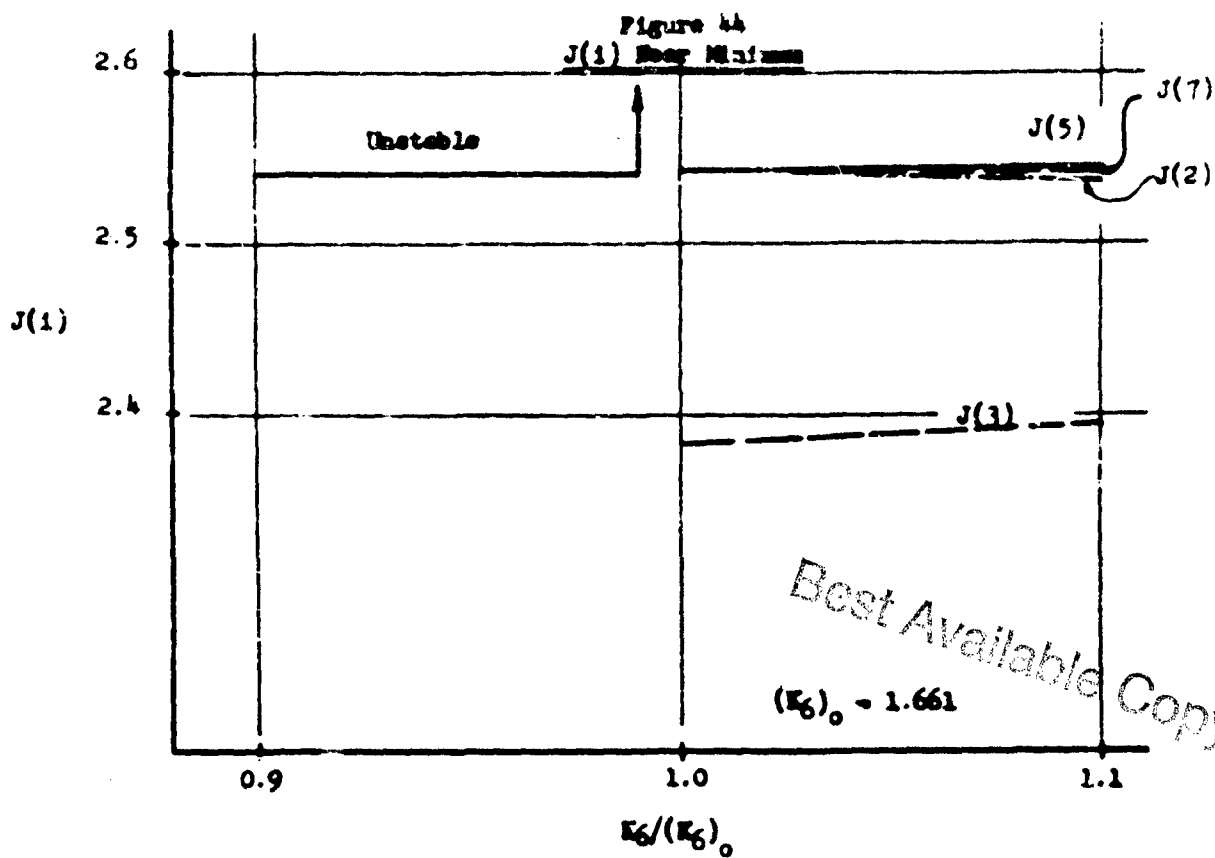
Figure 45

J(1) Near Minimum
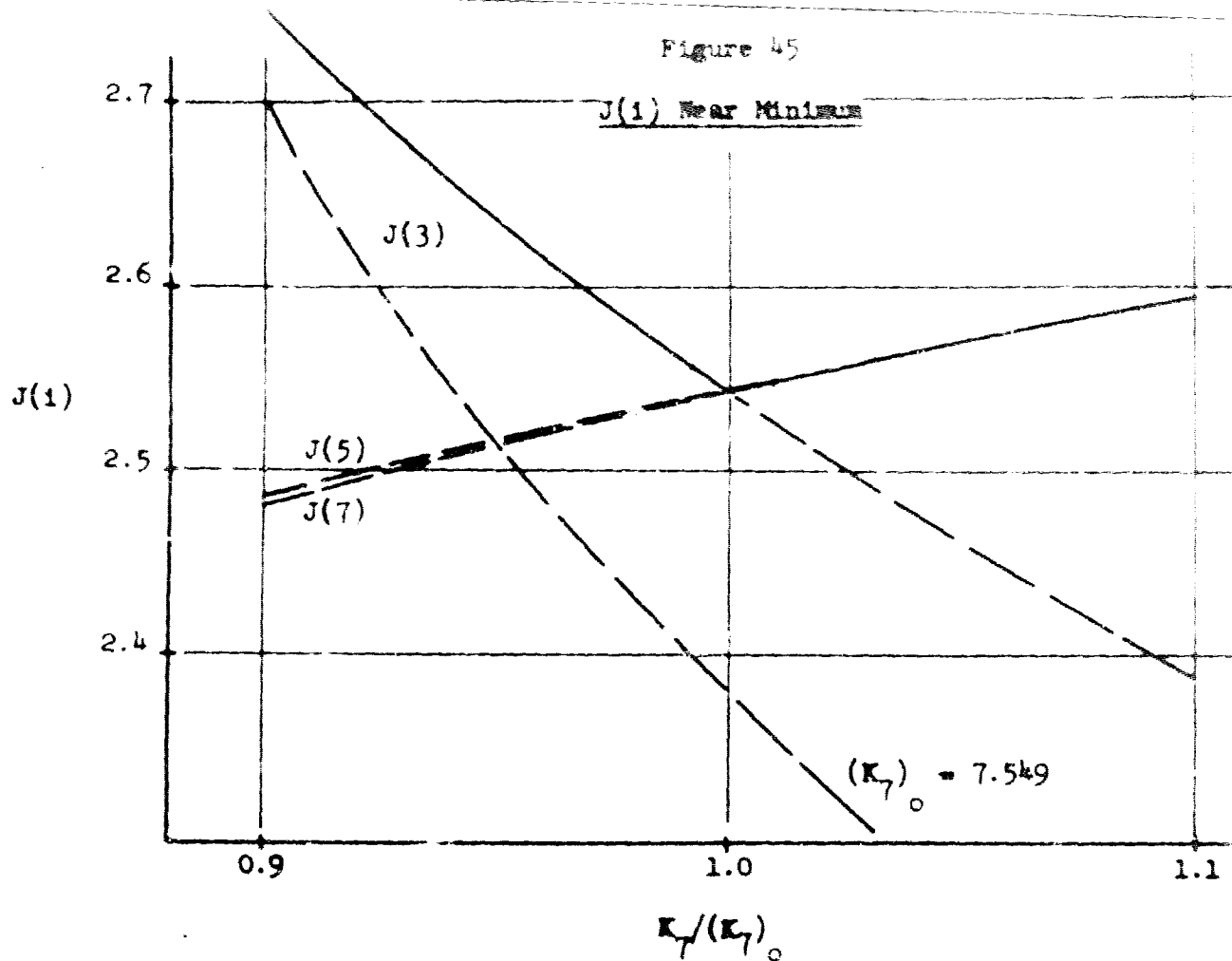
Figure 46
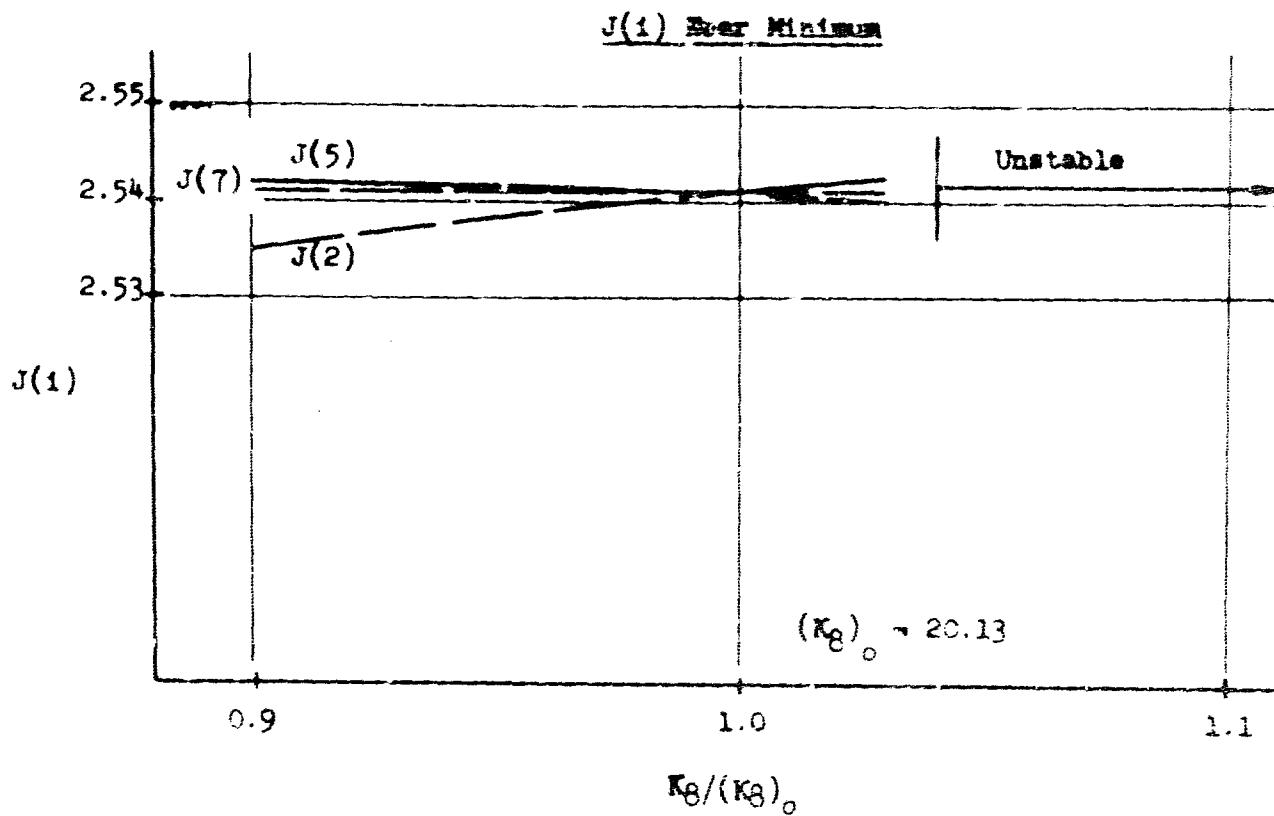
J(1) Near Minimum

Figure 47

J(1) Near Minimum
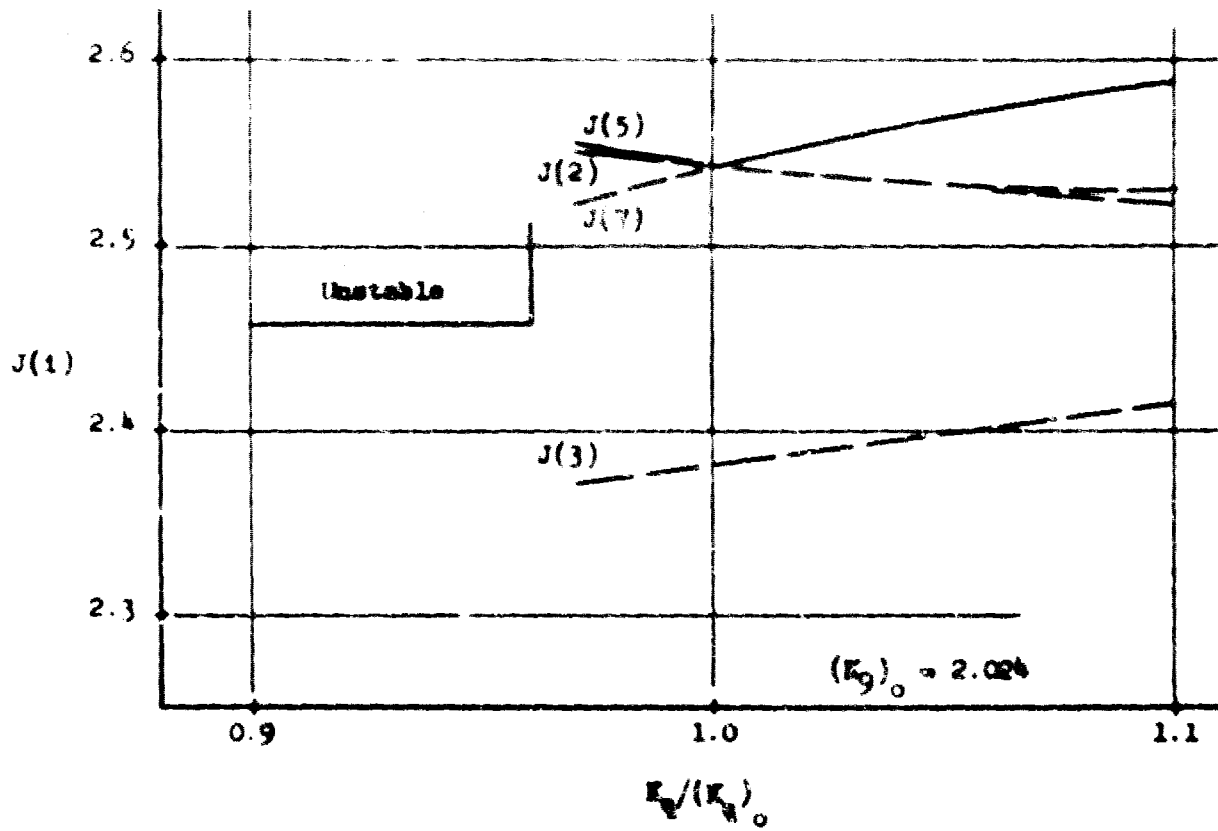
6.0 Conclusions and Recommendations

The Max-Ranking Criterion proves to be a simple but effective method for measuring and comparing the over-all performance of a system. It can be readily used as a performance index for systems subjected to either stochastic or deterministic disturbances. It also has the capability to be used as an index of performance for a wide variety of other systems optimization problems. It is impossible to prove that the Max-Ranking Criterion is the most general form of an optimization criterion. However, there is no criterion that the author has uncovered in the literature which is not a special case of this more general criterion.

The Max-Ranking Array can be constructed without any prior knowledge of the internal interactions of the system. The designer need only express his specification for as many measures of system response as he feels are important to the over-all system performance. If the designer's demands on the system are unrealistic, the results of the optimization study will point this out. These results can then be used to indicate where the specifications must be relaxed if a workable system is to be produced.

The one disadvantage of the Max-Ranking Criterion is that standard methods of calculus cannot, in general, be used to determine the optimum. For systems above 3rd and 5th order, which require several parameters to be determined for the optimization, this is not really a disadvantage at all. At this point, the algebraic complexities of the calculus approach become overwhelming, and other means of optimization are usually used.

The criterion can be readily used with either analog or digital computation; in fact, it requires less digital computation time than other multiple-parameter criteria which employ weighting functions.

The problem of determining an equivalent linearization gain does not appear to be difficult, at least for the case of the saturating element. Booton's method has proven to be acceptable for this type of nonlinearity as long as the limiting action is not severe. In cases where the nonlinearity cannot be handled in this fashion, digital computer simulation can be used.

On of the principal advantages of random search techniques is the ease with which they can be applied to large-scale, complex systems. At present, the search efficiency of the simpler methods could stand improvment. However, the efficiency can be improved if one is willing to accept more complicated search logic. Much more can be learned about the general behavior of random search techniques. Some questions which have not yet been answered, but which should be studied in the near future are:

1. How do unallowable (i.e., unstable) regions of various size affect the search efficiency of the single parameter search?

2. How does the probability of finding the true extremum of a multimodal function vary with the relative position and shape of the extrema?

3. How does multidimensionality affect the search effeciency?

4. Can large improvements in search efficiency be obtained without sacrificing too much programming simplicity?

These are only a few questions which need to be answered in the area of random search methods. This is certainly an area where very little practical information is available at this time; it is also one where modest efforts could lead to significant improvements in the general area of optimization.

Appendix A.1 References

1. Laning, J. H., Jr., and Battin, R. H., "Random Processes in Automatic Control," McGraw Hill Book Co., Inc., New York, 1956.

2. Crandall, S. H., Editor, "Random Vibration," M. I. T. Technology Press, Cambridge, Mass., 1958.

3. Lee, Y. K., "Statistical Theory of Communications," John Wiley and Sons, N. Y., 1960.

4. Wiener, N., "Extrapolation, Interpolation and Smoothing of Stationary Time Series," John Wiley, N. Y., 1949.

5. Sherman, Seymour, "Non-Mean Square Error Criteria," PGIT, IRE, Vol. IT-4, No. 3, Sept. 1958, pp. 125-126.

6. Newton, G. C., Jr., "Compensation of Feedback Control Systems Subject to Saturation," J. Franklin Institute, Vol. 254, No. 4, pp. 281-286, Oct. 1952; and Vol. 254, No. 5, pp. 391-413, Nov. 1952.

7. James, H. F., Nichols, N. B., and Phillips, R. S., "Theory of Servomechanisms," M. I. T. Radiation Lab. Series, Vol. 25, McGraw Hill Book Co., Inc., N. Y., 1947.

8. Hall, A. C., "The Analysis and Synthesis of Linear Servo Mechanisms," M. I. T. Press, Cambridge, Mass., 1943.

9. Zaborsky, J. and Diesel, J. W., "Probabilistic Error as a Measure of Control System Performance," Trans. AIEE, Vol. 78, pt. II, July 1959, pp. 163-168.

10. Milsum, J. H., "Problems in Optimizing Stochastically Disturbed Saturating Regulators Employing a Binary Error Criterion," ScD Thesis, Department of Mechanical Engineering, M. I. T. Cambridge, Mass., 1957.

11. Magdaleno, R. and Wolkovitch, J., "Performance Criteria for Linear Constant Coefficient Systems with Random Inputs," ASD-TDR 62-470, January 1963.

12. Gibson, J. E., et. al., "A Set of Standard Specifications for Linear Automatic Control Systems," Trans. AIEE, Vol. 80, pt. II, 1961, pp. 65-77.

13. Gibson, J. E., "Nonlinear Automatic Control," McGraw Hill, New York, 1963.

14. Wolkovitch, J., et. al., "Performance Criteria for Linear Constant Coefficient Systems with Deterministic Inputs," ASD Technical Report 6-501, December 1961.

15. Gille, J. C., Pelegrin, M. J., Decaulne, P., "Feedback Control Systems," McGraw Hill Book Co., Inc., N. Y., 1959.

16. Kazakov, I. E., "Approximate Probability Analysis of the Operational Precision of Essentially Nonlinear Feedback Control Systems," Automation and Remote Control, Vol. 18, No. 5, May 1956, pp. 423-450.

17. Booton, R. C., Jr., "Nonlinear Control Systems with Statistical Inputs," D.A.C.L. Report No. 61, M. I. T., Cambridge, Mass. 1952.

18. Booton, R. C., Jr., Mathews, M. V., and Seifert, W. W., "Nonlinear Servomechanisms with Random Inputs," D.A.C.L. Report No. 70, M. I. T., Cambridge, Mass., 1953.

19. This line of reasoning was developed by H. M. Paynter and was suggested to the author in a conversation.

20. Newton, G. C., Jr, Gould, L. A., and Kaiser, J. F., "Analytical Design of Linear Feedback Controls," John Wiley and Sons, New York, 1957.

21. Conversation with George C. Newton, Jr.

22. Crandall, S. H., "Engineering Analysis," McGraw Hill Book Co., Inc., New York, 1956.

23. Bellman, R., Introduction to Matrix Analysis, McGraw Hill, New York.

24. Paynter, H. M., and Suez, J., "Automatic Digital Setup and Scaling of Analog Computers," ISA Trans., Vol. 3, No. 1, Jan. 1964, pp. 55-64.

25. Box, G. E. P., and Wilson, K. B., "On the Experimental Attainment of Optimum Conditions," J. Royal Stat. Society, Series B, Vol. 13, 1951, pp. 1-45.

26. Brooks, S. H., "A Comparison of Maximum Seeking Methods," *Operations Research*, Vol. 7, 1959, pp. 430-457.

27. Gelfand, I. M., and Tsetlin, M. L., "The Principle of Non Local Search in Automatic Optimization Systems," *Soviet Phys. - Doklady*, Vol. 6, No. 3, Sept. 1961, pp. 192-194.

28. Brooks, S. H., "A Discussion of Random Methods for Surface Maxima," *Operations Research*, Vol. 6, 1958, pp 244-251.

29. Karnopp, D. C., "Search Theory Applied to Parameter Scan Optimization Problems," Ph.D. Thesis. Department of Mechanical Engineering, M. I. T. Cambridge, Mass., 1961.

30. Karnopp, D. C., "Random Search Techniques for Optimization Problems," *Automatica*, Vol. 1, No. 2/3, Pergamon Press, August 1963, pp 111-121.

31. Doob, J. L., "Stochastic Processes," John Wiley and Sons, Inc., New York, 1953.

32. Neumann, G., "An Ocean Wave Spectra and a New Method of Forecasting Wind Generated Sea," Beach Erosion Board, TM 43, Dec. 1954.

33. Pierson, J. D., "Water Waves," *Applied Mechanics Review*, V. 14, No. 1, January, 1961, pp. 1-3.

34. Pierson, W. J., Jr., "Wind Generated Gravity Waves," adv. in Geophysics, Vol. II, Academic Press, New York, 1955, pp. 93-178.

35. Broniwitz, L. E., "Amplitude-Limited Controllers for Systems with Random Disturbances," S.M. Thesis, Department of Electrical Eng., M. I. T., Cambridge, Mass., 1961.

36. Bass, R. W., discussion of a paper by A. M. Letov, Proc., Heidelberg, Conf. on Automatic Control, (Regelungstechnik: Moderne Theorien und ihre Verlvendbaikeit," by R. Oldenbourg, Munich, 1957), - pp 209-210.

37. Davis, M. C., "Analysis and Control of Ship Motions in a Random Seaway," S.M. Thesis, Department of Naval Architecture and Marine Engineering, M. I. T., June, 1961.

38. Kalman, R. E. and Bertram, J. E., "Control Systems Analysis and Design Via the Second Method of Lyapunov, I Continuous-Time Series," *Trans. of the ASME*, Series D., Jour. of Basic Engineering, Vol. 82, June 1960.

39. Korvin-Kroukovsky, B. V., "Theory of Seakeeping," SNAME, New York, 1961.

40. St. Denis, M., and Pierson, W. J., Jr., "On the Motions of Ships in Confused Seas," Trans. SNAME, 1953, p. 281.

41. Blagoveshensky, S. N., "Theory of Ship Motions," Vol. 1 and II, Dover Publications, 1962.

42. Newman, J. N., "A Linearized Theory for the Motion of a Thin Ship in Regular Waves," Jour. of Ship Research, Vol. 5, No. 1, June 1961, pp. 34-55.

43. Korvin-Kroukovsky, B. V., and Lewis, E. V., "Ship Motions in Regular and Irregular Seas," Inter. Shipbuilding Progress, Vol. 2, No. 6, 1955, pp 81-95.

44. Vossers, G., "Behavior of Ships in Waves," De Technische Uitgevery H. Stam N. V., Haarlem, The Netherlands, 1962.

45. Havelock, T. H., "The Forces on a Submerged Body Moving Under Waves," Part I, Motion Normal to the Wave Crests," Quart. Trans. of the Inst. of Naval Architects, Vol. 96, No. 2, April 1954, pp 77-81.

46. Milne-Thompson, L. M., "Theoretical Hydrodynamics," McMillan and Co., Ltd. London, Second Edition, 1949, -. 355.

47. Chadwick, J. H., Jr., "On the Stabilization of Roll," SNAME Trans. Vol. 63, 1955, pp. 237-280.

48. "Nomenclature for Treating the Motion of a Submerged Body through a Fluid," Technical and Research Bulletin No. 1-5, SNAME, 1952.

Appendix A.2 Digital Computer Programs

The Fortran listing for the three digital computer programs discussed in Chapter 5 are included here. A sufficient number of comment cards have been included to explain the logic and the mnemonic coding used.

```
COMPUTER PROGRAM FOR A THIRD ORDER SYSTEM WITH A THREE PARAMETER CONTROL.


      DIMENSION AKF(3),AKM(3),AK(3),CK0(3),CK1(3),CK2(3),RANK(3,10),
     1 SIGO(4),AJR(3)

      READ 10, NIR,NTEST,NSTOP,NLOOP,NEXP,ACC
C       NIR= NUMBER OF ROWS IN THE RANKING ARRAY
C       NTEST= NUMBER OF TIMES THE ENTIRE SEARCH IS TO BE CONDUCTED
C       NSTOP=NUMBER OF STABLE POINTS BEFORE TERMINATING EACH SEARCH
C       NLOOP= NUMBER OF UNSTABLE POINTS WHICH WILL TERMINATE THE SEARCH
C       NEXP= THE EXPONENT (M) IN THE SEARCH PROBABILITY DENSITY
C       ACC= THE MINIMAL ACCEPTABLE CHANGE WHICH CONSTITUTES AN IMPROVEMENT

      READ 20, AKO,OMEGAC,OMEGAN,ZETA
C       AKO= K(SUBSCRIPT)O
C       OMEGAC= OMEGA(SUBSCRIPT)C
C       OMEGAN= OMEGA(SUBSCRIPT)N        (THE NATURAL FREQUENCY)
C       ZETA= ZETA             (THE DAMPING RATIO)

      READ 20, (AKF(K),CK0(K),CK1(K),CK2(K),K=1,3)
C       AKF(K)= THE STARTING POINT FOR EACH SEARCH PARAMETER
C       CK0(K)= THE MULTIPLICATION FACTOR IN EACH SEARCH EQUATION
C       CK1(K)= THE LOWER LIMIT OF EACH SEARCH SPACE
C       CK2(K)= THE UPPER LIMIT OF EACH SEARCH SPACE

      READ 30,((RANK(I,J),I=1,3),J=1,NIR)
C       RANK(I,J)= THE RANKING ARRAY

   10 FORMAT(5I5,F10.0)
   20 FORMAT(4F10.0)
   30 FORMAT(3F10.0)

C       THE FIRST FEW NUMBERS GENERATED BY RANNOF(B) ARE NOT RANDOM, THUS IT
C            IS NECESSARY TO START THIS TYPE OF FUNCTION GENERATOR.
      DO 80 K=1,20
   80 G= RANNOF(B)

C       INITIAL PRINTOUT
      PRINT 100, AKO,OMEGAC,OMEGAN,ZETA
  100 FORMAT( 1H1 60H       TEST OF NEWTONS CONTROL FOR A SECOND ORDER SYS
     1TEM               //6X4HAKO=F10.3,3X7HOMEGAC=F10.3,3X7HOMEGAN=
     1E10.3,5X5HZETA= F10.3//)
      PRINT 110, (AKF(K),CK0(K),CK1(K),CK2(K), K=1,3)
  110 FORMAT(14X3HAKF17X3HCK017X3HCK117X3HCK2/(4F20.3))
      PRINT 120,((RANK(I,J),I=1,3),J=1,NIR)
  120 FORMAT(1H 4X11HRANKING ARRAY   /13X5HSIGOA15X5HSIGOB15X5HSIGOC
     1/ (3E20.3))
      PRINT 130, ACC,NTEST,NSTOP,NLOOP,NEXP
  130 FORMAT(1H 5X4HACC=F10.7, 4X6HNTEST=I10,4X6HNSTOP=I10,4X6HNLOOP=I10
     1,5X5HNEXP=I10)

C       CALCULATE THOSE CONSTANTS WHICH DO NOT VARY WITH AK(K).
      A10= 2.*ZETA
      A11= OMEGAN**2
      A12= 1./(AKO**2)
      A13= A11**2

C       SET UP THE NECESSARY TEST LOOPS
      DO 1000 ITEST= 1,NTEST
```
- A.6 -

```
C        SET TH  NECESSARY COUNTERS EQUAL TO ZERO
      N1=0
      N3=0
C      N1= THE TOTAL NUMBER OF POINTS TESTED
C      N2= THE TOTAL NUMBER OF STABLE POINTS TESTED

C      SET THE STARTING VALUE OF THE MINIMUM PERFORMANCE INDEX TO A HIGH NUMBER
      AJM= 99999.

C      PRINT TABLE HEADINGS
      PRINT 150
  150 FORMAT(1H01X2HN14X2HN911X2HAJ13X2HK113X2HK213X2HK312X5HSIGQA10X
     1 5HSIGQD10X5HSIGQF10X5HSIGQV)

C      SET THE STARTING POINT FOR THE SEARCH EQUAL TO THE PRESCRIB   VALUE
      DO 200 K=1,3
      AKM(K)= AKF(K)
  200 AK(K)= AKF(K)

C      USE THE GIVEN VALUES (AK(K)= AKF(K)   THE FIRST TIME THROUGH
      GO TO 400

C      RANDOM SEARCH ROUTINE
  300 DO 350 K=1,3
  310 AK(K)= CKQ(K)*(2.*RANNOF(R)-1.)**NFXP +AKM(K)
      IF(AK(K)-CK1(K))310,330,330
  330 IF(AK(K)-CK2(K)) 350,350,310
  350 CONTINUE
      N1= N1+1

C      TEST TO SEE IF TOO MANY UNSTABLE POINTS HAVE BEEN CHOSEN.  (THIS IS A
C            SAFETY SHUT-OFF)
      IF(N1-N3-NLOOP) 400,900,900

C      CALCULATE THOSE PARAMETERS WHICH DEPEND UPON AK(K)
  400 A2= 1.+AK(1)
      A1=(A10+AK(2))*OMEGAN
      A0=(1.+AK(3))*A11

C      CHECK STABILITY
      IF(A2/A0) 300,410,410
  410 IF(A1/A0) 300,411,411

C      CALCULATE THE MEAN SQUARE VALUES OF THE FOUR VARIABLES
  411 D3= A2
      D2= A1+A2*OMEGAC
      D1= A0+A1*OMEGAC
      D0=    A0*OMEGAC
      DELTA= 2.*D0*D3*(D1*D2-D0*D3)*A12
      SIGQ(1)= D0*D1/DELTA
      SIGQ(2)= A13*D2*D3/DELTA
      SIGQ(3)=(AK(1)**2*D0*D1+((OMEGAN*AK(2))**2-2.*AK(1)*A11*AK(3))
     1 *D0*D3+(A11*AK(3))**2*D2*D3)/DELTA
      SIGQ(4)= A11*D0*D3/DELTA
      N3= N3+1

C      HAVE ENOUGH STABLE POINTS BEEN TRIED
      IF(N3-NSTOP) 700,1000,1000

C      DETERMINE THE PERFORMANCE INDEX, AJ, FOR THIS POINT
  700 AJ= 0.
      DO 770 J= 1,3
      DO 720 IR= 2, NIR
      IF(RANK(J,IR) - SIGQ(J)) 720, 724, 724
```
                              - A.7 -

```
720 CONTINUE
    GO TO 300
724 AIR= IR
725 AJR(J)= AIR-2.+(SIGQ(J)-RANK(J,IR-1))/(RANK(J,IR)-RANK(J,IR-1))
    IF(AJ - AJR(J)) 765, 770, 770
765 AJ= AJR(J)
770 CONTINUE

C       IS AJ SUFFICIENTLY BETTER THAN THE PREVIOUS VALUE
    IF(AJ/AJM - ACC)  780,300,300

C       INDEX THE VALUES FOR THE NEW MINIMUM
780 DO 790 K=1,3
790 AKM(K)= AK(K)
    AJM= AJ

C       PRINT THE NEW MINIMUM
    PRINT 800, N1,N3,AJM,(AK(K),K=1,3),(SIGQ(J),J=1,4)
800 FORMAT(2I6,8F15.4)

C       RETURN TO SEE IF A NEW MINIMUM CAN BE FOUND
    GO TO 300
900 CONTINUE
1000 CONTINUE
    CALL EXIT
    END
```

```
      DIMENSION NAK(5),VK(5),CK0(5),CK1(5),CK2(5),AKF(5),AKM(5),AKN(5),
     1 RANK(4,10),ACC1(20),ML1(20),NFX(20),AJR(4),SIGX(4),SIGM(4),LK(5)
     1 , NSTOP(20)

      READ 20, NIR,NIA,NLOOP,N3L,NL
C         NIR= THE NUMBER OF ROWS IN THE RANKING ARRAY
C         NIA= THE NUMBER OF INDEPENDENT SEARCHES TO BE MADE
C         NLOOP= NUMBER OF UNSTABLE POINTS WHICH WILL TERMINATE THE SEARCH
C         N3L= NUMBER OF PURELY RANDOM POINTS TO BE TRIED
C         NL= NUMBER OF CONTROL PARAMETERS TO BE SEARCHED

      READ 22,        AKF(K),CK0(K),CK1(K),CK2(K), K= 1,5)
C       AKF(K)= THE STARTING POINT FOR EACH SEARCH PARAMETER
C       CK0(K)= THE MULTIPLICATION FACTOR IN EACH SEARCH EQUATION
C       CK1(K)= THE LOWER LIMIT OF EACH SEARCH SPACE
C       CK2(K)= THE UPPER LIMIT OF EACH SEARCH SPACE

      READ 24, (LK(L), L=1, NL)
C       LK(L)= THE CONTROL PARAMETERS TO BE SEARCHED

      READ 35, OMEGA, ZETA, EMASS
C       THE NATURAL FREQUENCY, DAMPING RATIO, AND MASS (OR MOMENT OF INERTIA)
C            OF THE SECOND ORDER SYSTEM

      READ 35, S0, ETA0, ETA1
C       THE INPUT FILTER PARAMETERS

      READ 40,((RANK(IC, IR), IC= 1, 4), IR= 1, NIR)
C       RANK(I,J)= THE RANKING ARRAY

      READ 45,(ACC1(IA),          NFX(IA),NSTOP(IA), IA= 1, NIA)
C       ACC= THE MINIMAL ACCEPTABLE CHANGE WHICH CONSTITUTES AN IMPROVEMENT
C       NEX= THE EXPONENT (M) IN THE SEARCH PROBABILITY DENSITY
C       NSTOP= NUMBER OF STABLE POINTS WHICH WILL TERMINATE THE SEARCH

   20 FORMAT (5I5)
   22 FORMAT(I5, 4F10.0)
   24 FORMAT (5I3)
   35 FORMAT(3E10.3)
   40 FORMAT(4E10.3)
   45 FORMAT(1F10.0, 3I5)

C       THE FIRST FEW NUMBERS GENERATED BY RANNOF(R) ARE NOT RANDOM, THUS IT
C            IS NECESSARY TO START THIS TYPE OF FUNCTION GENERATOR.
      DO 80 J= 1,20
   80 G= RANNOF(R)

C       INITIAL PRINTOUT
      PRINT 100
  100 FORMAT (1H156X19H0 A GALL - COURSE 2//49X41HOPTIMUM CONTROL - FIVE
     1 PARAMETER FEEDBACK/46X40HRANDOM SEARCH SOLUTION - MIN-MAX OPTIMUM
     1          )
      PRINT 110, ((RANK(IC, IR), IC= 1, 4), IR= 1, NIR)
  110 FORMAT(1H055X20HRANK-WEIGHTING ARRAY/93X4HSIGA16X4HSIGD16X4HSIGR
     116X4HSIGF/ (F40.3, 3F20.3))
      PRINT 115
```

- A.9 -

```
115 FORMAT(1H0)
    PRINT 140, OMEGA, ZETA, EMASS, SO, ETA0, ETA1
140 FORMAT (4X6HOMEGA=E10.3, 5X5HZETA=F10.3, 5X5HMASS= F10.3, 7X3HS0=
   1 F10.3, 3X7HETA(0)=F10.3, 3X7HETA(1)=F10.3)
    PRINT 150,(NAK(K),AKF(K),CK0(K),CK1(K),CK2(K), K=1,5)
150 FORMAT(1H0,8X12HSEARCH ARRAY  /22X6HNAK(K)14X6HAKF(K)14X6HCK0(K)
   1 14X6HCK1(K)14X6HCK2(K) / 5(20X15, 4F20.3/))
    PRINT 115

C       CALCULATE COEFFICIENTS WHICH DO NOT DEPEND UPON AK(K)
    CA1= 2. * ZETA * OMEGA
    D50= ETA1 + CA1
    D5A= OMEGA **2
    D4A= ETA1
    D3A= ETA0 **2
    D40= D3A + CA1 * ETA1 + D5A
    D30= CA1 * D3A + ETA1 * D5A
    D20= D3A * D5A
    CA= ETA0
200 DO 1100 IA= 1, NIA
    PRINT 210, ACC1(IA), NL1(IA), NEX(IA), NSTOP(IA), N3L
210 FORMAT(5X5HACC1=F12.6,4X4HNL1=I4,11X5HFXP.=I2,12X6HNSTOP=I9,11X4HN
   13L=I5)

C       SET INITIAL CONDITIONS
250 AJM= 99999.
    N1= 0
    N3=0

C       RANDOM SEARCH  (A PURELY RANDOM SEARCH IS CONDUCTED FOR THE FIRST N3L
C          TIMES - CONTROL PARAMETERS TO BE SELECTED CAN BE VARIED BY CHOOSING
C          LK(K)          )
    DO 260 K= 1,5
    AK(K) = AKF(K)
260 AKM(K)= AKF(K)
    NFXP= 1
    GO TO 370
300 IF(N3 - N3L) 301, 302, 320
301 NFXP= 1
    GO TO 320
302 NEXP= NEX(IA)
320 DO 330 L= 1, NL
    K= LK(L)
323 G= RANNOF(B)
    AK(K)= CK0(K) * (2. * G - 1.) **NFXP * AKM(K)
    IF(AK(K) - CK1(K)) 323, 329, 329
329 IF(AK(K) - CK2(K)) 330, 330, 323
330 CONTINUE

C       HAVE TOO MANY UNSTABLE POINTS BEEN FOUND
    IF(N1 - N3 - NLOOP) 370, 1050, 1050
370 N1= N1 + 1
C       HAVE ENOUGH STABLE POINTS BEEN CONSIDERED
371 IF(N3 - NSTOP(IA))  400, 1000, 1000

C       CALCULATE DENOMINATOR
400 D5= D50 + (1. + D5A * AK(1))* AK(5)
    D4= D40 + (D50 + D5A * (D4A * AK(1) + AK(2))) * AK(5)
    D3= D30 + (D40 + D5A * (D3A * AK(1) + D4A * AK(2) + AK(3)))* AK(5)
    D2= D20 + (D30 + D5A * (D3A * AK(2) + D4A * AK(3) + AK(4)))* AK(5)
```

- A.10 -

```
      D1= (D20 + D3A * (D3A * AK(5) + D4A * AK(4))) * AK(5)
      D0= D20 * AK(4) * AK(5)

C        CHECK STABILITY
      DA= D4 - D3 / D5
      IF(DA) 300, 300, 421
  421 DB= D2 - D1 / D5
      DC= D3 - D5 * DF / DA
      IF(DC) 300, 300, 424
  424 DD= D1 - D0 * D5 / DA
      DE= DB - DA * DD / DC
      IF(DE) 300, 300, 427
  427 DF= DD - D0 * DC / DE
      IF(DF) 300, 300, 429
  429 IF(D5) 300, 300, 440
  440 N3= N3 + 1

C        CALCULATE MEAN SQUARE VALUES
      FM1= -D0 * D1 * D5 + D0 * D3 **2 + D1 **2 * D4 - D1 * D2 * D3
      FM2= D0 * D3 * D5 + D1 **2 - D1 * D2 * D5
      FM3= D0 * D5 **2 + D1 * D3 - D1 * D4 * D5
      FM4= D2 * D5 **2 - D3 * D4 * D5 - D1 * D4 + D3 **2
      FMA5= D2 * FM4 - D4 * FM3 + FM2
      FM0= D4 * FM1 - D2 * FM2 + D0 * FM3
      DELTA= D0 * ( - D5 * FM4 + D5 * FM3) + D1 * FMA5
  500 CB= D3A + AK(5) **2
      CC= D3A * AK(5) **2
      CD= ETA0 + AK(5)
      CE= ETA0 * AK(5)
      D80= S0 / (2. * DELTA)
      D81= D80 / FMASS **2
      D82= D80 * D3A **2 * AK(5) **2
      SIGX(1)= D81 * (FM0 + CB * FM1 + CC * FM2)
      J=1
      GO TO 700
  521 SIGX(2) = D81 * (FM2 + CB * FM3 + CC * FM4)
      J=2
      GO TO 700
  531 C4= CA*AK(1) + AK(2)
      C3= CA*AK(2) + AK(3)
      C2= CA*AK(3) + AK(4)
      C1= CA*AK(4)
      C1A= C4**2 -2.*C3*AK(1)
      C1B= C3**2 -2.*C2*C4+2.*C1*AK(1)
      C1C= C2**2 -2.*C1*C3
      SIGX(3)= D82*(AK(1))**2 *(FM0+C1A*FM1+C1B*FM2+C1C*FM4+C1**2 *FM4)
      J=3
      GO TO 700
  541 SIGX(4)= D82*(AK(1))**2 *(FM1+C1A*FM2+C1B*FM3+C1C*FM4+CA**2*AK(4)*
     1 FMA5/(D20*AK(5)))
      J=4

C        CALCULATE EACH AJ FOR THIS POINT
  700 DO 720 IR= 2, NIR
      IF(RANK(J,IR) - SIGX(J)) 720, 724, 724
  720 CONTINUE
      GO TO 300
  724 AIR= IR
  725 AJR(J)= AIR-2.+(SIGX(J)-RANK(J,IR-1))/(RANK(J,IR)-RANK(J,IR-1))
      IF(AJM- AJR(J)) 300, 300, 730
  730 GO TO (521, 531, 541, 750), J
```

- A.11 -

```
  750 AJ= 0.
      DO 770 J= 1,4
      IF(AJ - AJR(J)) 765, 770, 770
  765 AJ= AJR(J)
  770 CONTINUE

C         IS AJ SUFFICIENTLY BETTER THAN THE PREVIOUS VALUE
      IF(AJ/AJM - ACC1(IA)) 800, 900, 900

C         IF A NEW MINIMUM, CALCULATE THE REMAINING MEAN SQUARE VALUES
  800 SIGX(5)= DB1 * (FM1 + CB * FM2 + CC * FM3)
      C4= CD * AK(1) + AK(2)
      C3= CE * AK(1) + CD * AK(2) + AK(3)
      C2= CE * AK(2) + CD * AK(3) + AK(4)
      C1= CE * AK(3) + CD * AK(4)
      C0= CE * AK(4)
      SIGX(6)= (DB1/AK(3)**2)*(AK(1)**2 *FM0+(C4**2 -2.*C3*AK(1))*FM1+(
     1 C3**2 -2.*C2*C4+2.*C1*AK(1))*FM2+(C4**2-2.*C1*C3+2.*C0*C4)*FM3+(
     1 C1**2-2.*C0*C2)*FM4+C0*CE*FM5/(D20)AK(3)))

C         STORE  THE VALUES FOR THE NEW MINIMUM
      DO 820 K=1, 5
  820 AKM(K)= AK(K)
      DO 825 J= 1, 6
  825 SIGM(J)= SIGX(J)
      AJM= AJ

C         PRINT OUT NEW MINIMUM AND THEN GO BACK AND SEARCH AGAIN
      PRINT 901, (AKM(K),K=1,5),AJM,(SIGM(J),J=1,6)
  901 FORMAT(7X3HKA=F12.5, 5X3HKV=F12.5, 5X3HKD=F12.5, 5X3HKF=F12.5,5X
     1 3HKP=E12.5, 5X3HAJ=E12.5/4X6HSIGXA=F12.5, 2X6HSIGXD=F12.5, 2X
     1 6HSIGXP=E12.5, 2X6HSIGXF=F12.5, 2X6HSIGXV=F12.5, 2X6HSIGXI=F12.5)
      GO TO 900
 1000 GO TO 1100
 1050 PRINT 1051, MLOOP
 1051 FORMAT(5X14,45H DISCARDED VALUES ABOVE THE ALLOWABLE MAXIMUM)
 1100 CONTINUE
 1101 CALL EXIT
      END
```

```
*     LIST
      DIMENSION C(10,12,2),A(100),AC(10),SIRK(12),RANK(8,10),AJR(9),
     1 B(100), DEL(100),W(100,8),WA(100),M(100),D(15),AK(9),AKM(9),
     1 AD(11),AKF(9),CK0(9),CK1(9),CK2(9),NFX(10),ACC(10),NL3(10),LK(9),
     1 SIGM(12)
      COMMON K,ETA21,ETA22,AJM,C,A,SIGV,RANK,AJR,AC,ACC,DEL,WN,V,WA,M,AD
     1,NIR,IA

C     DATA INPUT
      READ 10, NL, NIA, NIR, NLOOP, NSL,MM,NAL,NEXPO
C       NL= THE NUMBER OF CONTROL PARAMETERS TO BE SEARCHED
C       NIA= THE TOTAL NUMBER OF COMPLETE SEARCHES TO BE CONDUCTED
C       NIR= NUMBER OF ROWS IN THE RANKING ARRAY
C       NLOOP= NUMBER OF UNSTABLE POINTS WHICH WILL TERMINATE THE SEARCH
C       NSL= THE NUMBER OF PURELY RANDOM CHOICES TO BE MADE
C       MM= THE NUMBER OF FREQUENCIES USED IN FINITE DIFFERENCE INTEGRATION
C       NAL= THE NUMBER OF SEARCHES TO BE CONDUCTED WITH EXPONENT NEXPO
C       NEXPO= THE SECOND OF THREE EXPONENTS WHICH CAN BE CHOSEN FOR THE SEARCH

      READ 11, (LK(I), I=1,NL)
C       LK(I)= THE CONTROL PARAMETERS WHICH WILL BE SEARCHED

      READ 16, AMASS,AIY,ZW,ZWD,AMW,AMWD,ZQ,ZQD,AMQ,AMQD,RZR,UO,ALR,ALP
C       THESE ARE THE SYSTEM PARAMETERS WHICH ARE DEFINED IN SECTION 5.2.2

      READ 12,ETA21,ETA11,ETA01,ETA22,ETA12,ETA06
C       THESE ARE THE INPUT FILTER PARAMETERS DEFINED IN SECTION 5.2.2

      READ 13,(AKF(I),CK0(I),CK1(I),CK2(I), I= 1,9)
C       AKF(K)= THE STARTING POINT FOR EACH SEARCH PARAMETER
C       CK0(K)= THE MULTIPLICATION FACTOR IN EACH SEARCH EQUATION
C       CK1(K)= THE LOWER LIMIT OF EACH SEARCH SPACE
C       CK2(K)= THE UPPER LIMIT OF EACH SEARCH SPACE

      READ 17,((RANK(IC,IR), IC= 1,8), IR= 1, NIR)
C       RANK(I,J)= THE RANKING ARRAY

      READ 15, (ACC(IA),NL3(IA),NFX(IA), IA= 1,NIA)
C       ACC= THE MINIMAL ACCEPTABLE CHANGE WHICH CONSTITUTES AN IMPROVEMENT
C       NL3= THE TOTAL NUMBER OF STABLE SEARCHES TO BE CONDUCTED
C       NEX= THE EXPONENT (M) IN THE SEARCH PROBABILITY DENSITY

C       WA(M)= THE FREQUENCIES TO BE USED IN THE FINITE DIFFERENCE INTEGRATION
      READ 14, (WA(M), M=1,MM)

   10 FORMAT(8I10)
   11 FORMAT(9I3)
   12 FORMAT(6F10.3 )
   13 FORMAT(4F10.3)
   14 FORMAT(8F10.0)
   15 FORMAT(F10.0, 2I10)
   16 FORMAT(7E10.3)
   17 FORMAT(8F10.3)
```

- A-13 -

```
C          THE FIRST FEW NUMBERS GENERATED BY RANNOF(R) ARE NOT RANDOM, THUS IT
C             IS NECESSARY TO START THIS TYPE OF FUNCTION GENERATOR.
      DO 80 K=1,20
   80 G= RANNOF(R)

C     INITIAL PRINTOUT
      PRINT 100
  100 FORMAT(1H1,56X19H0 A GALL - COURSE 2//49X41HOPTIMUM CONTROL - NINE
     1PARAMETER FEEDBACK/48X40HRANDOM SEARCH SOLUTION - MIN-MAX OPTIMUM)
      PRINT 110, ((RANK(IC,IR), IC= 1,8), IR= 1, NIR)
  110 FORMAT(1H 55X20HRANK WEIGHTING ARRAY/15X115H SIGTA          SIGTA
     1      SIGZF         SIGZD         SIGTD         SIGSP        SIGSF
     1 SIGRP         SIGRF         /(6X8F15.4))
      PRINT 120, AMASS,AIY,ZW,ZWD,AMW,AMWD,ZQ,ZQD,AMQ,AMQD,RZR,UO,ALR,
     1 ALS,ETA21,ETA11,ETA01,ETA22,ETA14,ETA04
  120 FORMAT(1H0,10X9HMA SS=F15.3,7X9HIY=F10.3,7X9HZW=F10.3,4X4HZWD=F10.3,
     17X3HMW=E10.3,6X4HMWD=F10.3,/19X3HZQ=F10.3,6X4HZQD=F10.3,7X3HMQ=
     1E10.3,6X4HMQD=F10.3,6X4HRZR=F10.3,7X3H O=E10.3,/19X3HLR=F10.3,
     17X3HLS=E10.3,/10X6HETA21=F10.3,4X6HETA11=E10.3,4X4HETA01=F10.3,4X
     16HETA22 E10.3,4X6HETA14=E10.3,4X6HETA04=F10.3)

C          CALCULATE THOSE PARAMETERS WHICH DO NOT DEPEND UPON AK(K)
  150 A(1)= 1./(AMASS - ZWD)
      A(2)= A(1) * ZW
      A(5)= 1. / (AIY - AMQD)
      A(30)= A(5) * AMW
      A(31)= A(5) * AMWD
      A(40)= A(1) * (ZQ + AMASS * UO)
      A(41)= A(1) * ZQD
      A(60)= A(5) * RZR
      A(61)= A(5) * AMQ
      A(70)= 1. - A(31) * A(41)
      A(71)= A(30)*A(41)+A(31)*A(40)+A(61)+A(2)
      A(72)= A(30)*A(40)+A(60)-A(2)*A(61)
      A(73)= A(2)*A(60)
      D(11)= A(70)

C          IF D(11) IS NEGATIVE, THE STABILITY CHECKS LATER ON IN THE PROGRAM ARE
C             INCORRECT  -  THUS THE PROGRAM WILL BE STOPPED HERE
      IF(D(11)) 155, 155, 160
  155 PRINT 156
  156 FORMAT(10X50H011 IS NEGATIVE, MAKING THIS PROGRAM INVALID         )
      GO TO 1103

C          CONTINUE CALCULATING COEFFICIENTS WHICH DO NOT DEPEND UPON AK(K)
  160 A(3)= ETA01**2
      A(4)= ETA04**2
      A(6)= ETA14
      A(7)= A(4)
      A(8)= 0.
      A(9)= ETA11
      A(10)= ETA11 + ETA14
      A(11)= A(3) + A(4) + ETA11*ETA14
      A(12)= A(3) * ETA14 + A(4) * ETA11
      A(13)= A(3) * A(4)
      A(14)= A(3)
      A(15)= 0.
      A(16)= ETA14
      A(17)= A(4)
      A(19)= ETA11
      A(20)= A(3)
```

- A.14 -

```
      C(7,1,1)= 1.
      C(7,1,2 )= A(4)
      C(6,11,1)= 1.
      C(6,11,2)= A(4)
      C(7,2,1)= A(3)
      C(7,2,2)= 1.
      C(6,12,1)= A(3)
      C(6,12,2)= 1.
      C(5,4,1)= A(3)
      C(5,4,2)= 1.
      C(5,3,1)= -1.
      C(5,3,2)= -A(4)
      ETA21= ETA21*A(1)
      ETA22= ETA22*A(5)

C        CALCULATE AND STORE THE POWERS OF WA(M) WHICH ARE USED LATER ON
      DO 170 M=1,MN
      G= -WA(M)**2
      DO 170 N=2,6
  170 W(M,N)= G**(N-1)

C        SET INITIAL CONDITIONS
      DO 1000 IA= 1, NIA
      IA=IA
      PRINT 180, ACC(IA),NL3(IA),NEX(IA)
  180 FORMAT(1H05X4HACC=F10.7,6X4HNL3=I6,8X5HEXP.=I2///)
      AJM= 99999.
      N2= 0
      N3= 0
      DO 200 I= 1,9
      AK(I)= AKF(I)
  200 AKM(I)= AKF(I)

C        RANDOM SEARCH   -  THE FIRST TIME THROUGH, THE INITIAL VALUES ARE USED.
C           THE SECOND SERIES OF CHOICES CAN BE PURELY RANDOM.  THE THIRD SERIES
C           WILL HAVE THE EXPONENT NEXPO.  THE FINAL GROUP WILL HAVE THE EXPONENT
C           MEX.   THE LENGTH OF EACH GROUP CAN BE VARIED.
      NEXP= 1
      GO TO 332
C        CHOOSE AK(I)
  300 IF(N3 - N3L) 301, 302, 303
  301 NEXP= 1
      GO TO 320
  302 NEXP= NEXPO
      GO TO 320
  303 IF(N3 - N4L) 320, 304, 320
  304 NEXP= NEX(IA)
  320 DO 330 L= 1, NL
      I= LK(L)
  323 G= RANNOF(B)
      AK(I)= CK0(I)*(2. *R-1.)**NEXP + AKM(I)
      IF(AK(I) - CK1(I)) 329, 320, 320
  329 IF(AK(I) - CK2(I)) 330, 330, 323
  330 CONTINUE

C        HAVE TOO MANY UNSTABLE POINTS BEEN DETERMINED
      IF(N2 - N3 - MLCOP) 332, 900, 900
  332 N2= N2 + 1

C        CALCULATE THOSE PARAMETERS WHICH DEPEND ON AK(K) AND WHICH ARE NEEDED
C           FOR STABILITY CHECKING
```

- A.15 -

```
400 B(10)= -AK(1)
    B(11)= -AK(1) * AK(9)
    B(2)= -AK(1) * AK(2)
    B(70)= AK(8) + AK(9)
    B(71)= AK(8) * AK(9)
    B(20)= AK(4) * AK(8)
    B(21)= AK(9) * AK(9)
    B(30)= (-AK(4) + AK(9)) * A(3) + B(71)
    B(31)= (-B(20) + B(21)) * A(3)
    B(40)= U0 * AK(1)
    B(60)= (-ALS * AK(4) - ALR * AK(9)) * A(5) + B(71)
    B(61)= (-ALS * B(20) - ALR * B(21)) *A(5)
    A(72)= B(31) + A(41) * B(61)
    A(73)= B(61) + A(31) * B(31)
    B(74)= B(30) + A(40)*B(61) + A(41) * B(60) - A(41) * B(31)
    B(75)= B(60) + A(31) * B(30) + A(30) * B(31) - A(2) * B(61)
    B(76)= A(61) * B(30) + A(60) * B(31) - A(40) * B(60)
    B(77)= A(30)* B(30) - A(2) * B(60)
    B(70)= A(60) * B(30)

C      CALCULATE THE DENOMINATOR AND CHECK STABILITY
       D(10)= B(73) + A(70) -B(2) * B(72) - AK(6) * B(73)- A(71)
       IF(D(10)) 320, 320, 410
 410 D(9)= B(71)*A(70)-B(70)*A(71)-B(11)*B(72)-AK(7)*B(73)-B(2)*B(74)
    1 -AK(6)*B(75)-A(72)
       D(8)=-B(71)*A(71)-B(70)*A(72)-B(11)*B(74)-AK(7)*B(75)-B(10)*B(72)
    1 -B(73)+B(2)*B(76)-AK(6)*B(77)+B(2)*B(78)
       DA= D(9) - D(11)*B(8)/D(10)
       IF(DA) 320, 320, 411
 411 D(7)=-B(71)*A(72)+B(70)*A(73)-B(10)*B(74)-B(75)+B(11)*B(76)-B(40)
    1+B(73)-AK(7)*B(77)+B(2)*B(78)
       D(6)= B(71)*A(73)-B(40)*B(75)+B(10)*B(76)-B(77)+B(11)*B(78)
       DB= D(7) - D(6)*D(11)/D(10)
       DC= D(8) - DB*D(10)/DA
       IF(DC) 320, 320, 412
 412 D(5)= -B(40)*B(77) + B(10)*B(78)
       IF(D(5)) 320, 320, 413
 413 DD= D(6) - D(5)*D(10)/DA
       DE= DB - DD*DA/DC
       IF(DE) 320, 320, 414
 414 DF= DD - D(5)*DC/DE
       IF(DF) 320, 320, 415
 415 N3= N3 + 1

C      HAVE ENOUGH STABLE POINTS BEEN EXAMINED
       IF(N3 - NL3((A)) 419, 700, 700

C      CALCULATE THE DENOMINATOR AT EACH FREQUENCY, WA(M)
 419 DO 420 I=1, 11
 420 AD(I)=D(I)+A(10)*D(I-1)+A(11)*D(I+2)+A(12)*D(I+3)+A(13)*D(I+4)
       DO 430 M=1, MN
       DEL1= AD(1) + AD(3)*W(M,2)+AD(5)*W(M,3)+AD(7)*W(M,4)+AD(9)*W(M,5)
    1 +AD(11)*W(M,6)
       DEL2= WA(M)*(AD(2)+AD(4)*W(M,2)+AD(6)*W(M,3)+AD(8)*W(M,4)+AD(10)*
    1 W(M,5))
 430 DEL(M)= DEL1**2 + DEL2**2

C      CALCULATE THE SEVERAL MEAN SQUARE VALUES OF THE SYSTEM VARIABLES
 439 C(6,1,1)= B(70)-AK(6)*B(61)-A(61)
       C(5,1,1)= -A(61)*B(70)+B(71)-AK(6)*B(60)-AK(7)*A(1)
       C(4,1,1)= -A(60)*B(70)-A(61)*B(71)
                                        - A.16 -
```

```
      C(3,1,1)= -A(60)*R(71)-R(40)-R(40)*R(41)
      C(2,1,1)= -B(40)*B(60)
      C(6,1,2)= A(41)*B(70)+A(40)+AK(6)*R(9)
      C(5,1,2)= A(40)*B(70)+A(41)*R(71)+AK(6)*R(30)+R(31)*AK(7)
      C(4,1,2)= A(40)*R(71)+AK(7)*R(30)+R(9)
      C(3,1,2)= B(30)+R(31)*B(40)
      C(2,1,2)= B(30)*R(40)
      K=1
      CALL SIGMA
      IF(K) 800, 300, 444
  444 C(6,2,1)= A(31)*R(70)+A(30)+R(2)*R(61)
      C(5,2,1)= A(30)*R(70)+A(31)*R(71)+R(2)+R(60)+R(11)*R(61)
      C(4,2,1)= A(30)*R(71)+R(10)*R(41)+R(11)*R(60)
      C(3,2,1)= B(10)*B(60)
      C(6,2,2)= B(70)-R(2)*R(31)-A(2)
      C(5,2,2)= -A(2)*R(70)+B(71)-R(11)*R(91)-R(  *R(30)
      C(4,2,2)= -A(2)*R(71)-R(10)*R(31)-R(11)*R( )
      C(3,2,2)= -R(10)*R(30)
      K=2
      CALL SIGMA
      IF(K) 800, 300, 449
  449 DO 450 I=1,4
      C(I,3,1)= U0*C(I+3,2,1) - C(I+2,1,1)
  450 C(I,3,2)= L0*C(I+3,2,2  - C(I+2,1,2)
      K=3
      CALL SIGMA
      IF(K) 800, 300, 454
  454 DO 455 I= 1,4
      C(I,4,1)= C(I+2,2,1)
  455 C(I,4,2)= C(I+2,2,2)
      K=4
      CALL SIGMA
      IF(K) 800, 300, 458

C        CACULATE MORE PARAMETERS WHICH VARY WITH AK(K)
  458 B(90)= B(2) + A(31)*AK(6)
      B(91)= A(30)*AK(6)+R(11)-A(61)*R(2)+A(31)*AK(7)
      B(92)= -AK(11)-A(60)*B(2)+A(30)*AK(7)+A(31)-A(6)*R(11)
      B(93)= A(30)+A(31)*B(40)-A(60)*R(11)+A(61)*AK(1)
      B(94)= A(30)*B(40)-A(60)*AK(1)
      B(95)= A(41)*B(2)+AK(6)
      B(96)= A(40)*B(2)+A(41)*B(11)+AK(7)-A(2)*AK(6)
      B(97)= A(40)*R(11)-A(41)*AK(1)+1.-A(2)*AK(7)
      B(98)= -A(40)*AK(1)+B(40)-A(2)
      B(99)= -A(2)*R(40)

C        CALCULATE MORE MEAN SQUARE VALUES
  459 C(7,5,1)=B(40)*R(20)
      C(7,5,2)= B(95)  *B(20)
      DO 460 I= 3,6
      I1= 96 - I
      C(I,5,1)=(AK(9)*R(I1) + B(I1+1)   )*R(20)
      I1= 101 - I
  460 C(I,5,2)=(AK(9)*R(I1) + B(I1+1) )*R(20)
      C(2,5,1)= AK(9)*R(94)  *B(20)
      C(2,5,2)= AK(9)*R(99)   *R(20)
      K=5
      CALL SIGMA
      IF(K) 800, 300, 464
```

- A.17 -

```
  464 DO 465 I=1,6
      C(I,6,1)= C(I+1,5,1)
  465 C(I,6,2)= C(I+1,5,2)
      K=6
      CALL SIGMA
      IF(K) 800, 300, 469
  469 C(7,7,1)= R(90)      *R(21)
      C(7,7,2)= R(95)      *B(23)
      DO 470 I= 3,6
      I1= 96 - I
      C(I,7,1)=(AK(8)*R(I1)+R(I1+1)      )*R(21)
      I1= 101 - I
  470 C(I,7,2)=(AK(8)*R(I1)+R(I1+1)      )*R(21)
      C(2,7,1)= AK(8)*R(94)  *B(21)
      C(2,7,2)= AK(8)*R(99)      *R(21)
      K=7
      CALL SIGMA
      IF(K) 800, 300, 474
  474 DO 475 I=1,6
      C(I,8,1)= C(I+1,7,1)
  475 C(I,8,2)= C(I+1,7,2)
      K=8
      CALL SIGMA
      IF(K) 800, 300, 479

C        THE REMAINING MEAN SQUARE VALUES ARE CALCULATED ONLY IF A NEW MINIMUM
C        HAS BEEN FOUND
  479 C(7,9,1)= AK(2)
      C(7,9,2)= AK(2)*A(4)
      DO 480 I=1,6
      C(I,9,1)= AK(3)*C(I+1,1,1) + AK(2)*C(I,1,1) - C(I,3,1)
  480 C(I,9,1)= AK(3)*C(I+1,1,2) + AK(2)*C(I,1,2) - C(I,3,2)
      K=9
      CALL SIGMA
  481 C(7,10,1)= B(90)
      C(6,10,1)= R(70)*R(90)+R(91)
      C(5,10,1)= R(71)*R(90)+R(70)*R(91)+R(92)
      C(4,10,1)= R(71)*R(91)+R(70)*R(92)+R(93)
      C(3,10,1)= R(71)*R(92)+R(70)*R(93)+R(94)
      C(2,10,1)= R(71)*R(93)+R(70)*R(94)
      C(1,10,1)= R(71)*R(94)
      C(7,10,2)= B(95)
      C(6,10,2)= R(70)*R(95)+R(96)
      C(5,10,2)= R(71)*B(95)+R(70)*B(96)+B(97)
      C(4,10,2)= B(71)*B(96)+R(70)*R(97)+R(98)
      C(3,10,2)= B(71)*R(97)+R(70)*R(98)+R(99)
      C(2,10,2)= R(71)*R(98)+R(70)*R(99)
      C(1,10,2)= R(71)*R(99)
      K=10
      CALL SIGMA
  489 DO 490 I=1,5
      C(7,11,1)= C(I+1,1,1)
  490 C(I,11,2)= C(I+1,1,2)
      K=11
      CALL SIGMA
  494 DO 495 I=2,5
      C(I,12,1)= C(I+1,2,1)
  495 C(7,12,2)= C(I+1,2,2)
      K=12
      CALL SIGMA
```

- A.18 -

```
C          STORE THE VALUES FOR THE NEW MINIMUM
      DO 550 I=1,12
  550 SIGM(I)= SIGX(I)
      AJ=0
      DO 560 J=1,8
      IF(AJR(J) - AJ) 560, 560, 555
  555 AJ= AJR(J)
  560 CONTINUE
      AJM= AJ
      DO 570 I= 1,9
  570 AKM(I)= AK(I)

C          PRINT THE VALUES FOR THE NEW MINIMUM
      PRINT 600, N9, AJM, (AK(I), I= 1,9)
  600 FORMAT(5X3HN9= I8, 5X4HAJM= F12.4,/5X7HK(1-9)=  5X9F12.4)
      PRINT 610, N1, (SIGM(I), I= 1,12)
  610 FORMAT(  3HN1=I6,3X6HSIGZA=F10.4,4X6HSIGTA=E10.4,4X6HSIGZF=F10.4,
     1 .X6HSIGTD=E10.4,4X6HSIGSP=E10.4,4X6HSIGSF=E10.4,/14X6HSIGPP=E10.4,
     1 .X3HSIGDF=E10.4,4X6HSIG21=E10.4,4X6HSIGT1=F10.4,4X6HSIGZV=E10.4,
     14X6HSIGTV=E10.4)
      PRINT 620, (AJR(I), I= 1,8)
  620 FORMAT(5X10HAJR(1-8)          8F14.4//)

C          CONTINUE THE SEARCH
      GO TO 300
  700 CONTINUE

C          FINAL PRINTOUT  - USED MOSTLY FOR DEBUGGING
  800 PRINT 801, (A(I), I= 1, 100)
      PRINT 801, (B(I), I= 1, 100)
      PRINT 801,(((C(I1,I2,I3), I1=1,10),I2=1,12),I3=1,2)
      PRINT 801, (D(I), I=1,15)
      PRINT 801, DA,DB,DC,DD,DE,DF
      PRINT 801, (AD(I), I= 1,11)
      PRINT 801, (AC(I), I=1,10)
      PRINT 801, (SIGX(I), I=1,12)
      PRINT 801, (AK(I), I=1,9)
      PRINT 801, (WA(K), K=1, MM)
      PRINT 801, ((W(M,N), M=1,MM), N=2,6)
  801 FORMAT(1H0/(10F13.5))
      PRINT 802, AJM, K
  802 FORMAT(1H0/ E20.5, I5          )
      GO TO 1000
  900 PRINT 901, NLOOP
  901 FORMAT(1H0010X16,40HDISCARDED VALUES.  SEARCH STOPPED.           )
      GO TO 700
 1000 CONTINUE
 1100 CALL EXIT
      END
```

```
C        SUBROUTINE USED WITH THIS PROGRAM
      SUBROUTINE SIGMA
*     LIST
      DIMENSION C(10,12,2),A(100),AC(10),SIGX(12),RANK(8,10),AJR(8),
     1 ACC(10),DFL(100),W(100,6),WA(100),H(100)         ,AD(11)
      COMMON K,ETA21,ETA22,AJM,C,A,SIGX,RANK,AJR,AC,ACC,DFL,MN,W,WA,H,AD
     1,NIR,IA

C        CALCULATE THE MEAN SQUARE VALUE IN EACH CASE
      DO 100 I= 1, 7
  100 AC(I+3)=    ETA21*(C(I,K,1)+A(6)*C(I+1,K,1)+A(7)*C(I+2,K,1)
     1            )+ETA22*(C(I,K,2)+A(9)*C(I+1,K,2)+A(14)*C(I+2,K,2)      )
      AC(2)= ETA21*A(7)*C(1,K,1)+ETA22*A(14)*C(1,K,2)
      AC(3)=    ETA21*(                A(7)*C(2,K,1)+A(6)*C(1,K,1))+ETA22*
     1(            A(14)*C(2,K,2)+A(9)*C(1,K,2))
      DO 200 M=1,MN
      C1=        AC(3)*W(M,2)+AC(5)*W(M,3)+AC(7)*W(M,4)+AC(9)*W(M,5)
      C2= WA(M)*(AC(2)+AC(4)*W(M,2)+AC(6)*W(M,3)+AC(8)*W(M,4)+AC(10)*
     1 W(M,5))
  200 H(M)= (C1**2+C2**2)/DFL(M)
      SIGX(K)= H(1)*WA(1)
      DO 210 M=2,MN
  210 SIGX(K)= SIGX(K)+(H(M-1)+H(M))*(WA(M)-WA(M-1))
      SIGX(K)= SIGX(K) /3.14159

C        CALCULATE AJ FOR THE FIRST 8 MEAN SQUARE VALUES
  250 IF(K-8) 260, 260, 400
  260 DO 300 IR= 2, NIR
      IR=IR
      IF(RANK(K,IR) - SIGX(K)) 300, 310, 310
  300 CONTINUE
  301 K=0
      GO TO 400
  310 AIR= IR
      AJR(K)= AIR-2.+(SIGX(K)-RANK(K,IR-1))/(RANK(K,IR)-RANK(K,IR-1))

C        IF THIS VALUE OF AJ IS LESS THAN THE PREVIOUS MINIMUM, CONTINUE THE
C           CALCULATIONS, OTHERWISE DETERMINE A NEW SET OF CONTROL PARAMETERS
      IF(AJR(K)/AJM - ACC(IA)) 400, 301, 301
  400 RETURN
      END
```